ZiLOG

Digital Signal Processors

Includes Specifications for the following parts:

Z89C00
Z89120/920
Z89121/921
Z89320/321

Product Specifications Databook

ZILOG

Digital Signal Processors

Includes Specifications for the following parts:

- **Z89C00**
- **Z89120/920**
- **Z89121/921**
- **Z89320/321**

Databook

DC 8299-01

DSP DATABOOK

TABLE OF CONTENTS

INTRODUCTION

Zilog's Focus on Application Specific Products Helps You Maintain Your Technological Edge

Zilog's DSP products are suitable for a broad range of applications, from general-purpose use through speech synthesis and mass storage. Whichever device you choose, you'll find a comprehensive feature set and easy-to-use development tools to speed your design time to production.

Z89C00 16-Bit Digital Signal Processor

With a high-performance single-cycle multiply/accumulate instruction and zero software overhead pointer architecture, the Z89C00 is an excellent choice for many DSP designs. Flexible general-purpose I/O features, including a 16-bit address and data bus and a 16-bit I/O bus, make it easy to configure even slow peripherals into the system. A comprehensive set of development support tools makes it even easier to work with the Z89C00. This device offers broad functionality in an affordable package for consumer and industrial product designs alike.

Z89120, Z89920 (ROMless) 16-Bit Mixed Signal Processor

Multiple-chip capability in a single-chip solution is the hallmark of the Z89120. Combining 16-bit DSP functions with an integrated 8-bit microcontroller and A/D, D/A converters, it is an optimal choice for communications applications including audio, fax, voice mail, modems and data transmission, as the Z89120 can handle several of these functions without additional hardware. Its very low power consumption and small footprint make it ideal for portable use, or in applications requiring long-term continuous operation, such as security systems and other supervisory instrumentation. The Z89920 is the ROMless version of the Z89120 device.

Z89121, Z89921 (ROMless) 16-Bit Mixed Signal Processor

The Z89121 system processor offers exceptional flexibility for applications like voice mail and personal communications, which involve substantial I/O and storage requirements. Two Codec ports allow extensive analog interfacing, and expanded DSP program memory space—plus a 32-Mbit DRAM interface—accommodates digital speech generation and storage. The Z8912's compact design provides maximum space-efficiency for remote messaging and paging applications. The Z89921 is the ROMless version of the Z89121 device.

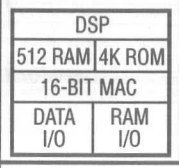
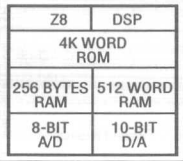
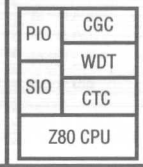
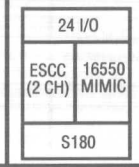
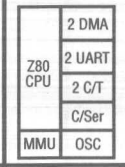
Z89320 16-Bit Digital Signal Processor

The Z89320 provides the computational power of the Z89C00 at a cost effective price. This device incorporates 512 bytes of RAM and 4K words of program ROM. Two general purpose user inputs and two user outputs provide convenient peripheral monitoring or control. A dedicated 16-bit I/O bus assists in transferring information to and from external peripherals. The compact instruction set is standard to all Zilog DSP products and provides ease-of-use programming. Applications include high-volume multimedia, digital audio, speech processing, and system control.

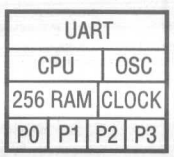
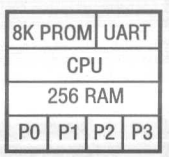
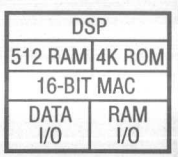
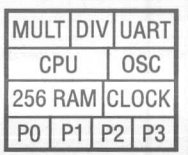
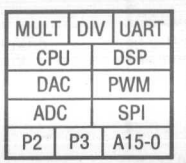
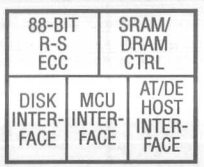
Z89321 16-Bit Digital Signal Processor

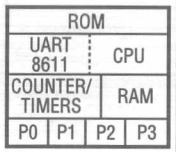
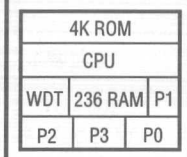
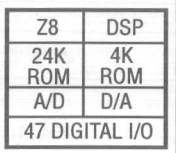
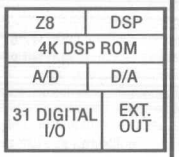
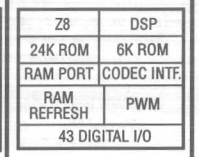
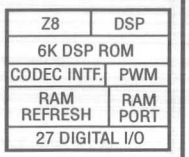
Building on the Z89320 feature set, the Z89321 integrates a dual codec interface to assist in the transfer of analog signals to the processor. A standard 8-bit codec can be used to communicate with the interface. An upgrade to the Z89321 will provide an expansion to the interface capabilities to include 16-bit linear and 16-bit stereo codecs. This integration path provides additional cost advantages to customers that use codecs for transfer of data.

Fax/Modem Superintegration™ Products Guide

	Data Pump	Single Chip		Controllers			
Block Diagram							
Part #	**Z89C00**	**Z89120**	**Z89920**	**Z84C15**	**Z80182**	**Z80180**	**Z85230**
Description	16-Bit Digital Signal Processor	Zilog Modem/Fax Controller (ZMFC)	Zilog Modem/Fax Controller (ZMFC)	IPC/EIPC Controller	Zilog Intelligent Peripheral (ZIP™)	High-performance Z80® CPU with peripherals	Enhanced Serial Com. Controller
Process/Speed	CMOS 10, 15 MHz	CMOS 20 MHz	CMOS 20 MHz	CMOS 6, 10,16 MHz	CMOS 16, 20 MHz	6, 8, 10, 16*, 20* *Z8S180 only	CMOS 8, 10,16, 20 MHz
Features	16-bit Mac 75 ns 2 data RAMs (256 words each) 4K word ROM 64Kx16 Ext. ROM 16-bit I/O Port 74 instructions Most single cycle Two conditional branch inputs, two user outputs Library of software macros available zero overhead pointers	Z8® controller with 24 Kbyte ROM 16-bit DSP with 4K word ROM 8-bit A/D 10-bit D/A (PWM) Library of software macros available 47 I/O pins Two comparators Independent Z8® and DSP Operations Power-Down Mode	Z8 w/64K external memory DSP w/4K word ROM 8-bit A/D 10-bit D/A Library of macros 47 I/O pins Two comparators Independent Z8® and DSP Operations Power-Down Mode	Z80® CPU, SIO, CTC WDT, CGC The Z80 Family in one device Power-On Reset Two chip selects 32-bit CRC WSG EV mode[1] 3 and 5 Volt Version	Complete Static Version of Z180™ plus ESCC (2 channels of Z85230) 16550 MIMIC 24 Parallel I/O Emulation Modes[1]	Enhanced Z80® CPU MMU 1 Mbyte 2 DMAs 2 UARTs with BRGs C/Serial I/O Port Oscillator Z8S180 includes; Pwr dwn, Prgmble EMI, divide-by-one clock option	Full dual-channel SCC plus deeper FIFOs: 4 bytes on Tx 8 bytes on Rx DPLL counter per channel Software compatible to SCC
Package	68-pin PLCC 60-pin VQFP	68-pin PLCC	68-pin PLCC	100-pin QFP 100-pin VQFP	100-pin QFP 100-pin VQFP	64-pin DIP 68-pin PLCC 80-pin QFP	40-pin DIP 44-pin PLCC
Other Applications	16-bit General-Purpose DSP TMS 32010/20/25 applications	Multimedia-Audio Voicemail Speech Storage and Transmission Modems FAXes, Sonabouys	Multimedia-Audio Voicemail Speech Storage and Transmission Modems FAXes, Sonabouys	Intelligent peripheral controllers Modems	General-Purpose Embedded Control Modem, Fax, Data Communications	Embedded Control	General-Purpose datacom. High performance SCC software compatible upgrade

Mass Storage Superintegration™ Products Guide

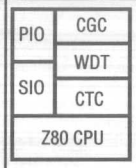
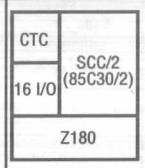
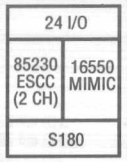
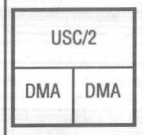
Block Diagram						
Part #	**Z86C91/Z8691**	**Z86E21**	**Z89C00**	**Z86C93**	**Z86C95**	**Z86018**
Description	ROMless Z8®	Z8® 8K OTP	16-Bit Digital Signal Processor	Enhanced Z8®	Enhanced Z8® with DSP	Zilog Datapath Controller (ZDPC)
Process/Speed	CMOS 16 MHz (C91) NMOS 12 MHz (91)	CMOS 12, 16 MHz	CMOS 10, 15 MHz	CMOS 20, 25 MHz	CMOS 24 MHz	CMOS 40 MHz
Features	Full duplex UART 2 Standby Modes (STOP and HALT) 2x8 bit Counter/Timer	8K OTP ROM 256 Byte RAM Full-duplex UART 2 Standby Modes (STOP and HALT) 2 Counter/Timers ROM Protect option RAM Protect option Low EMI option	16-bit Mac 75 ns 2 data RAMs (256 words each) 4K word ROM 64Kx16 Ext. ROM 16-bit I/O Port 74 instructions Most single cycle Two conditional branch inputs, two user outputs Library of software macros available zero overhead pointers	16x16 Multiply 1.7 µs 32x16 Divide 2.0 µs Full duplex UART 2 Standby Modes (STOP and HALT) 3 16-bit Counter/Timers Pin compatible to Z86C91 (PDIP)	8 channel 8-bit ADC, 8-bit DAC 16-bit Multiply/Divide Full duplex UART SPI (Serial Peripheral Interface) 3 Standby Modes (STOP/HALT/PAUSE) Pulse Width Modulator 3x16-bit timer 16-bit DSP slave processor 83 ns Mult./Accum.	Full track read Automatic data transfer (Point & Go®) 88-bit Reed Solomon ECC "on the fly" Full AT/IDE bus interface 64 KB SRAM buffer 1 MB DRAM buffer Split data field support 100-pin VQFP package JTAG boundary scan option Up to 8 KB buffer RAM reserved for MCU
Package	40-pin DIP 44-pin PLCC 44-pin QFP	40-pin DIP 44-pin PLCC 44-pin QFP	68-pin PLCC 60-pin VQFP	40-pin DIP 44-pin PLCC 44-pin QFP 48-pin VQFP	80-pin QFP 84-pin PLCC 100-pin VQFP	100-pin VQFP 100-pin QFP
Application	Disk Drives Modems Tape Drives	Software Debug Z8® prototyping Z8® production runs Card Reader	Disk Drives Tape Drives Servo Control Motor Control	Disk Drives Tape Drives Modems	Disk Drives Tape Drives Servo Control Motor Control	Hard Disk Drives

Block Diagram						
Part #	**Z08600/Z08611**	**Z86C30/E30 Z86C40/E40**	**Z89C65**	**Z89C66**	**Z89C67**	**Z89C68**
Description	Z8® NMOS (CCP™) 8600 = 2K ROM 8611 = 4K ROM	Z8® Consumer Controller Processor (CCP™) with 4K ROM C30 = 28-pin C40 = 40-pin E30/E40 = OTP version	Telephone Answering Controller with DSP LPC voice synthesis and DTMF detection	Telephone Answering Controller with DSP LPC voice synthesis and DTMF detection and external ROM/RAM interface	Telephone Answering Controller with digital voice encode and decode DTMF detection and full memory control interface	Telephone Answering Controller with digital voice encode and decode DTMF detection and external ROM/RAM interface
Process/Speed	NMOS 8,12 MHz	CMOS 12 MHz	CMOS 20 MHz	CMOS 20 MHz	CMOS 20 MHz	CMOS 20 MHz
Features	2K/4K ROM 128 Bytes RAM 22/32 I/O lines On-chip oscillator 2 Counter/Timers 6 vectored, priority interrupts UART (Z8611)	4K ROM, 236 RAM 2 Standby Modes 2 Counter/Timers ROM Protect RAM Protect 4 Ports (86C40/E40) 3 Ports (86C30/E30) Brown-Out Protection 2 Analog Comparators Low EMI Watch-Dog Timer Auto Power-On Reset Low Power option	Z8® Controller 24K ROM 16-bit DSP 4K Word ROM 8-bit A/D with AGC DTMF macro available LPC macro available 10-bit PWM D/A Other DSP software options available 47 I/O Pins	Z8® Controller 16-bit DSP 4K Word ROM 8-bit A/D with AGC DTMF macro available LPC macro available 10-bit PWM D/A Other DSP software options available External ROM/RAM capability 31 I/O Pins	Z8® Controller 24K ROM 16-bit DSP 6K Word ROM DTMF macro available LPC macro available 10-bit PWM D/A Other DSP software options available ARAM/DRAM/ROM Controller & Interface Dual Codec Interface 43 I/O	Z8® Controller 64K ROM (external) 16-bit DSP, 6K word ROM DTMF macro available LPC macro available 10-bit PWM D/A Other DSP software options available ARAM/DRAM control/ interface External ROM/RAM Dual Codec Interface 27 I/O
Package	28-pin DIP 40-pin DIP 44-pin PLCC	28-pin DIP 40-pin DIP 44-pin PLCC, QFP	68-pin PLCC	68-pin PLCC	84-pin PLCC	84-pin PLCC
Application	Low cost tape board TAD	Window Control Wiper Control Sunroof Control Security Systems TAD	Fully featured cassette answering machines with voice prompts and DTMF signaling	General-Purpose DSP applications in TAD and other high-performance 1-tape voice processors	Voice Processing, DSP applications in tapeless TAD and other high-performance 1-tape voice processors	Voice Processing, DSP applications in tapeless TAD and other high-performance 1-tape voice processors

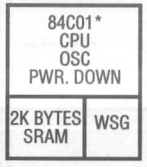
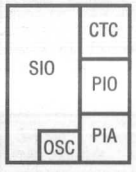
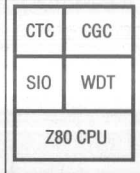
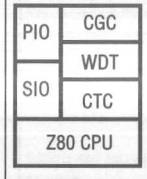
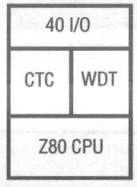
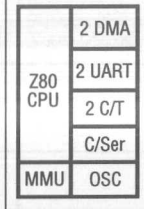
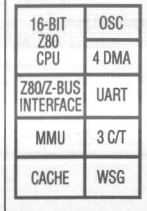
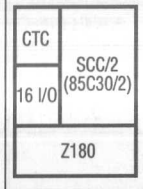
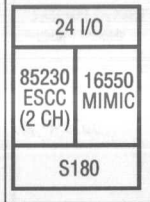
ZiLOG Video Products Superintegration™ Products Guide

	TV Controller			IR Controller		Cable TV	
Block Diagram	8K ROM 4K CHAR ROM Z8 CPU | RAM OSD 13 PWM | TIMER WDT | 5 PORTS	6K ROM 3K CHAR ROM Z8 CPU | RAM OSD 7 PWM | TIMER WDT | 3 PORTS	CHAR ROM COMMAND INTERPRETER ANALOG SYNC/DATA SLICER | OSD CTRL	1K/6K ROM Z8 CPU WDT | 124 RAM P2 | P3	2K/8K/16K ROM Z8 CPU WDT | 128,256, 768 RAM P0 | P1 | P2 | P3	4K ROM CPU WDT | 236 RAM | P1 P2 | P3 | P0	16K ROM | UART CPU | 236 RAM P0 | P1 | P2 P3 | P4 | P5 | P6
Part #	**Z86C27/127/97**	**Z86227**	**Z86128**	**Z86L06/L29**	**Z86L70/71/72** (Q193)	**Z86C40/E40**	**Z86C61/62**
Description	Z8® Digital Television Controller MCU with logic functions needed for Television Controller, VCRs and Cable	Standard DTC features with reduced ROM, RAM, PWM outputs for greater economy	Line 21 Controller (L21C™) for Closed Caption Television	18-pin Z8® Consumer Controller Processor (CCP™) low-voltage and low-current battery operation 1K-6K ROM	Z8® (CCP™) low-voltage parts that have more ROM, RAM and special Counter/Timers for automated output drive capabilities	Z8® Consumer Controller Processor (CCP™) with 4K ROM (C40) E40 = OTP version	Z8® MCU with Expanded I/O's and 16K ROM
Process/Speed	CMOS 4 MHz	CMOS 4 MHz	CMOS 12 MHz	Low Voltage CMOS 8 MHz	Low Voltage CMOS 8 MHz	CMOS 12 MHz	CMOS 16, 20 MHz
Features	Z8/DTC Architecture 8K ROM, 256-byte RAM 160x7-bit video RAM On-Screen Display (OSD) video controller Programmable color, size, position attributes 13 PWMs for D/A conversion 128-character set 4Kx6-bit char. Gen. ROM Watch-Dog Timer (WDT) Brown-Out Protection 5 Ports/36 pins 2 Standby Modes Low EMI Mode	Z8/DTC Architecture 6K ROM, 256-byte RAM 120x7-bit video RAM OSD on board Programmable color, size, position attributes 7 PWMs 96-character set 3Kx6-bit character generator ROM Watch-Dog Timer (WDT) Brown-Out Protection 3 Ports/20 pins 2 Standby Modes Low EMI Mode	Conforms to FCC Line 21 format Parallel or serial modes Stand-alone operation On-board data sync and slicer On-board character generator - Color - Blinking - Italic - Underline	Z8® Architecture 1K ROM & 6K ROM Watch-Dog Timer 2 Analog Comparators with output option 2 Standby Modes 2 Counter/Timers Auto Power-On Reset 2 volt operation RC OSC option Low Noise option Brown-Out Protection High current drivers (2, 4)	Z8® Architecture 2K/8K/16K ROM Watch-Dog Timer 2 Analog Comparators with output option 2 Standby Modes 2 Enhanced Counter/Timers, Auto Pulse Reception/Generation Auto Power-On Reset 2 volt operation RC OSC option Brown-Out Protection High current drivers (4)	4K ROM, 236 RAM 2 Standby Modes 2 Counter/Timers ROM Protect RAM Protect 4 Ports Brown-Out Protection 2 Analog Comparators Low EMI Watch-Dog Timer Auto Power-On Reset Low Power option	16K ROM Full duplex UART 2 Standby Modes (STOP and HALT) 2 Counter/Timers ROM Protect option RAM Protect option Pin compatible to Z86C21 C61 = 4 Ports C62 = 7 Ports
Package	64-pin DIP 52-pin active (127)	40-pin DIP	18-pin DIP	18-pin DIP 18-pin SOIC	20-pin DIP (L71), 18-pin DIP, SOIC (L70) 40,44-pin DIP, PLCC, QFP (L72)	40-pin DIP	40-pin DIP (C61) 44-pin PLCC,QFP (C61) 68-pin PLCC (C62)
Application	Low-end Television Cable/Satellite Receiver	Low-end Television Cable/Satellite Receiver	TVs, VCRs, Decoders	I.R. Controller Portable battery operations	I.R. Controller Portable battery operations	Window Control Wiper Control Sunroof Control Security Systems TAD	Cable Television Remote Control Security

ZiLOG Datacommunications Superintegration™ Products Guide

Block Diagram									
Part #	**Z8030/Z80C30 Z8530/Z85C30**	**Z85230/Z80230 Z85233***	**Z16C35**	**Z84C15**	**Z80181**	**Z80182**	**Z16C30**	**Z16C33**	**Z16C32**
Description	Serial Com. Controller	Enhanced Serial Com. Controller	Integrated Serial Com. Controller	Intelligent Peripheral Controller	Smart Access Controller	Zilog Intelligent Peripheral	Universal Serial Controller	Mono-channel Universal Serial Controller	Integrated Universal Serial Controller
Process/ Speed/ Clock Data Rate	NMOS: 4, 6, 8 MHZ CMOS: 8,10 16 MHz 2, 2.5, 4 Mb/s	CMOS: 10, 16 20 MHz 2.5, 4.0, 5.0 Mb/s	CMOS: 10, 16 MHz 2.5, 4.0 Mb/s	CMOS 6, 10,16 MHz	10, 12.5	CMOS 16, 20 MHz	CMOS: 20 MHz CPU Bus 10 Mb/s 20 Mb/s	CMOS: 10 MHz CPU Bus 10 Mb/s	CMOS:20 MHz CPU Bus 16 Mb/s 20 Mb/s
Features	Two independent full-duplex channels Enhanced DMA support: 10x19 status FIFO 14-bit byte counter NRZ/NRZI/FM	Full dual-channel SCC plus deeper FIFOs: 4 bytes on Tx 8 bytes on Rx DPLL counter per channel Software compatible to SCC *One channel of Z85230	Full dual-channel SCC plus 4 DMA controllers and a bus interface unit	Z80® CPU, SIO, CTC WDT, CGC The Z80 Family in one device Power-On Reset Two chip selects 32-bit CRC WSG EV mode[1] 3 and 5 Volt Version	Complete Z180™ plus SCC/2 CTC 16 I/O lines Emulation Mode[1]	Complete Static version of Z180 plus ESCC (2 channels of 85230) 16550 MIMIC 24 Parallel I/O Emulation Mode[1]	Two dual-channel 32-byte receive & transmit FIFOs 16-bit bus B/W: 18.2 Mb/s 2 BRGs per channel Flexible 8/16-bit bus interface	Single-channel (half of USC™) plus Time Slot Assigner functions for ISDN	Single-channel (half of USC) plus two DMA controllers Array chained and linked-list modes with ring buffer support
Package	40-pin DIP 44-pin CERDIP 44-pin PLCC	40-pin DIP 44-pin PLCC *44-pin QFP (85233)	68-pin PLCC	100-pin QFP 100-pin VQFP	100-pin QFP	100-pin QFP 100-pin VQFP	68-pin PLCC	68-pin PLCC	68-pin PLCC
Application	General-Purpose datacom.	General-Purpose datacom. High performance SCC software	High performance datacom. SCC upgrades	Intelligent peripheral controllers Modems	Intelligent peripheral controllers Printers, Faxes, Modems, Terminals	General-Purpose Embedded Control Modem, Fax, Data Communications	General-Purpose high-end datacom. Ethernet HDLC X.25 Frame Relay	General-Purpose high-end datacom. Ethernet HDLC X.25 Frame Relay	General-Purpose high-end datacom. Ethernet HDLC X.25 Frame Relay

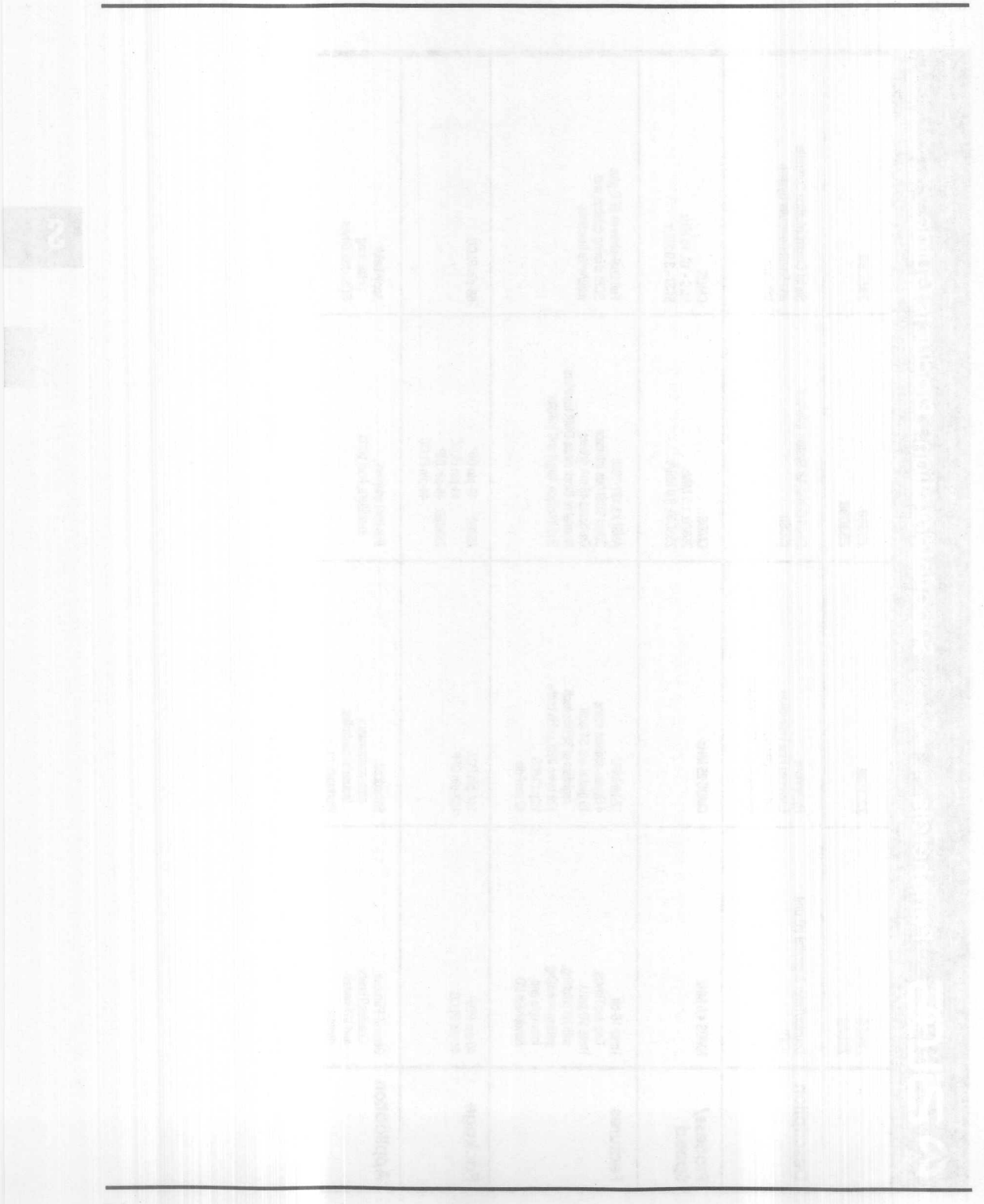
ZiLOG Z80® Embedded Controllers Superintegration™ Products Guide

Block Diagram									
Part #	**Z84C50**	**Z84C90**	**Z84013/C13**	**Z84015/C15**	**Z84011/C11**	**Z80180/S180**	**Z80280**	**Z80181**	**Z80182**
Description	Z80/84C01 with 2K SRAM	Killer I/O (3 Z80 peripherals)	Intelligent Peripheral Controller	Intelligent Peripheral Controller	Parallel I/O Controller	High-performance Z80® CPU with peripherals	16-bit Z80® code compatible CPU with peripherals	Smart Access Controller	Zilog Intelligent Peripheral
Speed MHz	10	8, 10, 12.5	6, 10	6, 10, 16	6, 10	6, 8, 10, 16*, 20* *Z8S180 only	10, 12	10, 12.5	16, 20
Features	Z80® CPU 2 Kbytes SRAM WSG Oscillator Pin compatible with Z84C00 DIP & PLCC EV mode[1] *84C01 is available as a separate part	SIO, PIO, CTC plus 8 I/O lines	Z80® CPU, SIO, CTC WDT, CGC, WSG, Power-On Reset 2 chip selects EV mode[1]	Z80® CPU, SIO, CTC WDT, CGC The Z80 Family in one device Power-On Reset Two chip selects 32-bit CRC WSG EV mode[1]	Z80® CPU, CTC, WDT 40 I/O lines bit programmable Power-On Reset EV mode[1]	Enhanced Z80 CPU MMU 1 Mbyte 2 DMAs 2 UARTs with BRGs C/Serial I/O Port Oscillator Z8S180 includes; Pwr dwn, Prgmble EMI, divide-by-one clock option	16-bit code compatible Z80® CPU Three stage pipeline MMU 16 Mbyte CACHE 256 byte Inst. & Data Peripherals 4 DMAs, UART, 3 16-bit C/T, WSG Z80/Z-BUS® interface	Complete Z180 plus SCC/2 CTC 16 I/O lines Emulation Mode[1]	Complete Static Version of Z180™ plus ESCC (2 channels of Z85230) 16550 MIMIC 24 Parallel I/O Emulation Modes[1]
Package	40-pin DIP 44-pin PLCC 44-pin QFP	84-pin PLCC	84-pin PLCC	100-pin QFP 100-pin VQFP	100-pin QFP	64-pin DIP 68-pin PLCC 80-pin QFP	68-pin PLCC	100-pin QFP	100-pin QFP 100-pin VQFP
Application	Embedded Controllers	General-purpose peripheral that can be used with Z80 and other CPU's	Intelligent datacom controllers	Intelligent peripheral controllers Modems	Intelligent parallel-I/O controllers Industrial display terminals	Embedded Control	Embedded Control Terminals Printers	Intelligent peripheral controllers Printers, Faxes, Modems, Terminals	General-Purpose Embedded Control Modem, Fax, Data Communications

[1] Allows use of existing development systems.

	Z8036 Z8536	Z32H00	Z5380 Z53C80	Z85C80
Description	Counter/Timer & parallel I/O Unit (CIO)	Hyperstone Enhanced Fast Instruction Set Computer (EFISC) Embedded (RISC) Processor	Small Computer System Interface (SCSI)	Serial Communication Controller and Small Computer System Interface
Process/ Speed	NMOS 4,6 MHz	CMOS 25 MHz	CMOS Z5380: 1.5 MB/s Z53C80: 3.0 MB/s	CMOS SCC - 10, 16 MHz SCSI - 3.0 MB/s
Features	Three 16-bit Counter/Timers, Three I/O ports with bit catching, pattern matching interrupts and handshake I/O	32-bit MPU 4 Gbytes address space 19 global and 64 local registers of 32 bits each 128 bytes instruction cache 1.2µ CMOS 42 mm² die	ANSI X3.131-1986 Direct SCSI bus interface On-board 48 mA drivers Normal or Block mode DMA transfers Bus interface, target and initiator	Full dual-channel SCC plus SCSI sharing databus and read/write functions
Package	40-pin PDIP 44-pin PLCC	144-pin PGA 132-pin QFP	Z5380: 40-pin DIP 44-pin PLCC Z53C80: 48-pin DIP 44-pin PLCC	68-pin PLCC
Application	General-Purpose Counter/Timers and I/O system designs	Embedded high-performance industrial controller Workstations	Bus host adapters, formatters, host ports	AppleTalk® networking SCSI disk drives

[2] Software and hardware compatible with discrete devices.



Z89C00

16-BIT DIGITAL SIGNAL PROCESSOR

FEATURES

- 16-Bit Single Cycle Instructions
- Zero Overhead Hardware Looping
- 16-Bit Data
- Ready Control for Slow Peripherals
- Single Cycle Multiply/Accumulate (100 ns)
- Six-Level Stack
- 512 Words of On-Chip RAM
- Static Single-Cycle Operation
- 16-Bit I/O Port
- 4K Words of On-Chip Masked ROM
- Three Vectored Interrupts
- 64K Words of External Program Address Space
- Two Conditional Branch Inputs/Two User Outputs
- 24-Bit ALU, Accumulator and Shifter
- IBM® PC Development Tools

GENERAL DESCRIPTION

The Z89C00 is a second generation, 16-bit, fractional, two's complement CMOS Digital Signal Processor (DSP). Most instructions, including multiply and accumulate, are accomplished in a single clock cycle. The processor contains 1 Kbyte of on-chip data RAM (two blocks of 256 16-bit words), 4K words of program ROM and 64K words of program memory addressing capability. Also, the processor features a 24-bit ALU, a 16 x 16 multiplier, a 24-bit Accumulator and a shifter. Additionally, the processor contains a six-level stack, three vectored interrupts and two inputs for conditional program jumps. Each RAM block contains a set of three pointers which may be incremented or decremented automatically to affect hardware looping without software overhead. The data RAMs can be simultaneously addressed and loaded to the multiplier for a true single cycle multiply.

There is a 16-bit address and a 16-bit data bus for external program memory and data, and a 16-bit I/O bus for transferring data. Additionally, there are two general purpose user inputs and two user outputs. Operation with slow peripherals is accomplished with a ready input pin. The clock may be stopped to conserve power.

Development tools for the IBM PC include a relocatable assembler, a linker loader, and an ANSI-C compiler. Also, the development tools include a simulator/debugger, a cross assembler for the TMS320 family assembly code and a hardware emulator.

To assist the user in understanding the Z89C00 DSP Q15 two's complement fractional multiplication, an application note has been included in this product specification as an appendix.

Notes:
All Signals with a preceding front slash, "/", are active Low, e.g., B/W (WORD is active Low); /B/W (BYTE is active Low, only).

Power connections follow conventional descriptions below:

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

GENERAL DESCRIPTION (Continued)

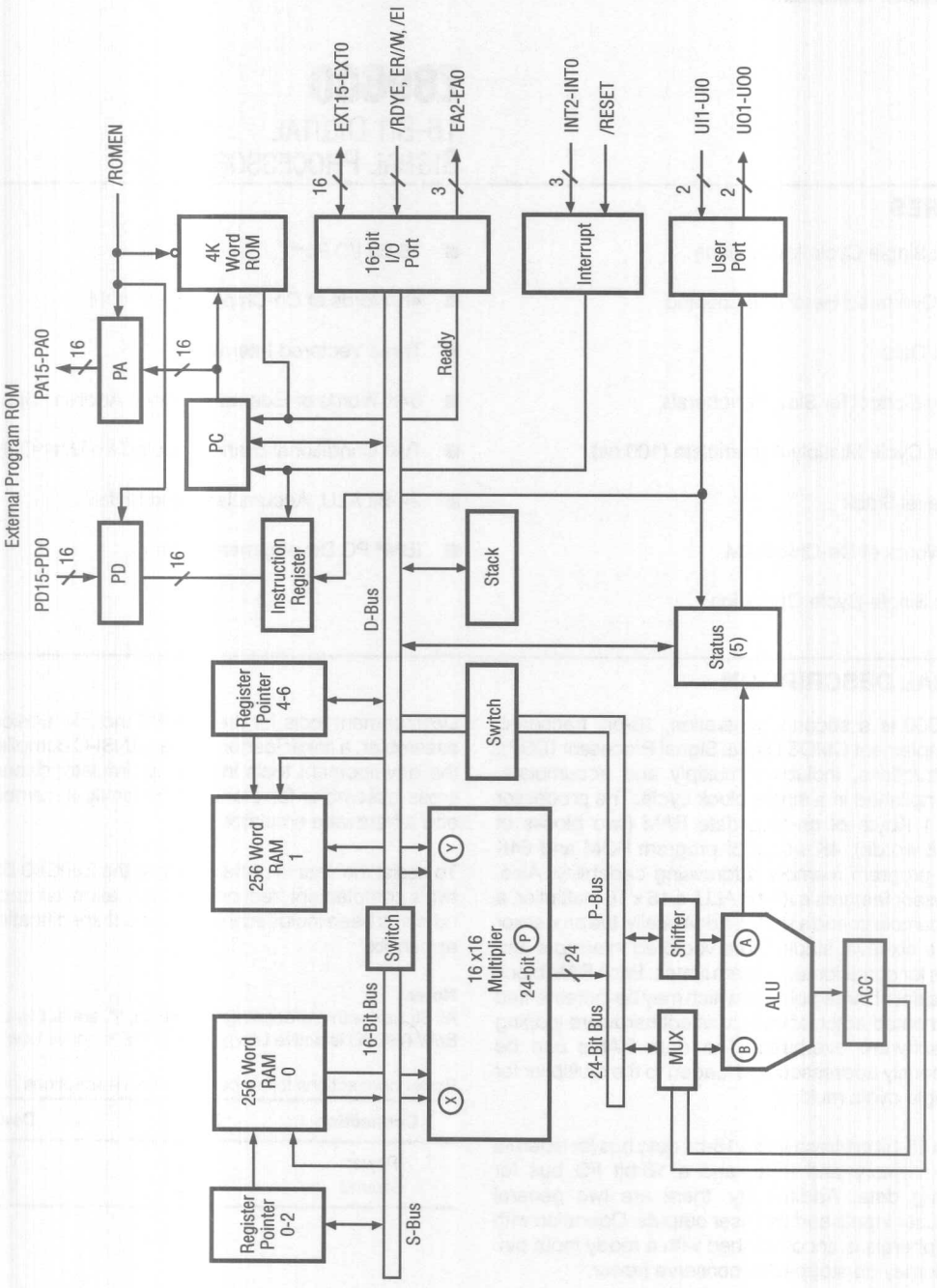


Figure 1. Functional Block Diagram

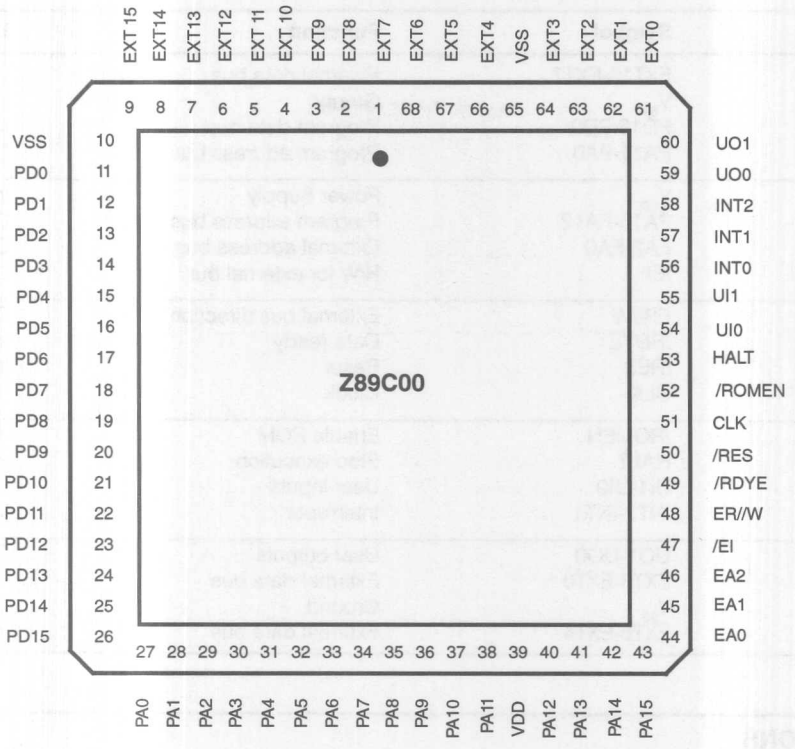


Figure 2. 68-Pin PLCC Pin Assignments

1

Table 1. 68-Pin PLCC Pin Identification

No.	Symbol	Function	Direction
1-9	EXT15-EXT7	External data bus	Input/Output
10	V_{SS}	Ground	Input
11-26	PD15-PD0	Program data bus	Input
27-38	PA11-PA0	Program address bus	Output
39	V_{DD}	Power Supply	Input
40-43	PA15-PA12	Program address bus	Output
44-46	EA2-EA0	External address bus	Output
47	/EI	R/W for external bus	Output
48	ER/W	External bus direction	Output
49	/RDYE	Data ready	Input
50	/RES	Reset	Input
51	CLK	Clock	Input
52	/ROMEN	Enable ROM	Input
53	HALT	Stop execution	Input
54-55	UI1-UI0	User inputs	Input
56-58	INT2-INT1	Interrupts	Input
59-60	UO1-UO0	User outputs	Output
61-64	EXT3-EXT0	External data bus	Input/Output
65	V_{SS}	Ground	Input
66-68	EXT6-EXT4	External data bus	Input/Output

PIN FUNCTIONS

CLK. *Clock* (input). External clock. The clock may be stopped to reduce power.

EXT15-EXT0. *External Data Bus* (input/output). Data bus for user defined outside registers such as an ADC or DAC. The pins are normally in output mode except when the outside registers are specified as source registers in the instructions. All the control signals exist to allow a read or a write through this bus.

ER/W. *External Bus Direction* (output, active Low). Data direction signal for EXT-Bus. Data is available from the CPU on EXT15-EXT0 when this signal is Low. EXT-Bus is in input mode (high-impedance) when this signal is High.

EA2-EA0. *External Address* (output). User-defined register address output. One of eight user-defined external registers is selected by the processor with these address pins for read or write operations. Since the addresses are part of the processor memory map, the processor is simply executing internal reads and writes.

/EI. *Enable Input* (output). Write timing signal for EXT-Bus. Data is read by the external peripheral on the rising edge of /EI. Data is read by the processor on the rising edge of CLK, not /EI.

HALT. *Halt State* (input). Stop Execution Control. The CPU continuously executes NOPs and the program counter remains at the same value when this pin is held High. This signal must be synchronized with CLK.

INT2-INT0. *Three Interrupts* (rising edge triggered). Interrupt request 2-0. Interrupts are generated on the rising edge of the input signal. Interrupt vectors for the interrupt service starting address are stored in the program memory locations 0FFDH for INT0, 0FFEH for INT1 and 0FFFH for INT2. Priority is : 2 = lowest, 0 = highest.

PA15-PA0. *Program memory address bus* (output). For up to 64K x 16 external program memory. These lines are tri-stated during Reset Low.

PD15-PD0. *Program Memory Data Input* (input). Instructions or data are read from the address specified by PD15-PD0, through these pins and are executed or stored.

/RES. *Reset* (input, active Low). Asynchronous reset signal. A Low level on this pin generates an internal reset signal. The /RES signal must be kept Low for at least one clock cycle. The CPU pushes the contents of the PC onto the stack and then fetches a new Program Counter (PC) value from program memory address 0FFCH after the Reset signal is released. RES Low tri-states the PA and PD bases.

/ROMEN. *ROM Enable* (input). An active Low signal enables the internal ROM. Program execution begins at 0000H from the ROM. An active High input disables the ROM and external fetches occur from address 0000H.

/RDYE. *Data Ready* (input). User-supplied Data Ready signal for data to and from external data bus. This pin stretches the /EI and ER/W lines and maintains data on the address bus and data bus. The ready signal is sampled from the rising edge of the clock with appropriate setup and hold times. The normal write cycle will continue from the next rising clock only if ready is active.

UI1-UI0. *Two Input Pins* (input). General purpose input pins. These input pins are directly tested by the conditional branch instructions. These are asynchronous input signals that have no special clock synchronization requirements.

UO1-UO0. *Two Output Pins* (output). General purpose output pins. These pins reflect the inverted value of status register bits S5 and S6. These bits may be used to output data by writing to the status register.

1

ADDRESS SPACE

Program Memory. Programs of up to 4K words can be masked into internal ROM. Four locations are dedicated to the vector address for the three interrupts (0FFDH-0FFFH) and the starting address following a Reset (0FFCH). Internal ROM is mapped from 0000H to 0FFFH, and the highest location for program is 0FFBH. If the /ROMEN pin is held high, the internal ROM is inactive and the processor executes external fetches from 0000H to FFFFH. In this case, locations FFFC-FFFF are used for vector addresses.

Internal Data RAM. The Z89C00 has an internal 512 x 16-bit word data RAM organized as two banks of 256 x 16-bit words each, referred to as RAM0 and RAM1. Each data RAM bank is addressed by three pointers, referred to as Pn:0 (n = 0-2) for RAM0 and Pn:1 (n = 0-2) for RAM1. The RAM addresses for RAM0 and RAM1 are arranged from 0-255 and 256-511 respectively. The address pointers, which may be written to or read from, are 8-bit registers connected to the lower byte of the internal 16-bit D-Bus and are used to perform no overhead looping. Three addressing modes are available to access the Data RAM: register indirect, direct addressing, and short form direct. These modes are discussed in detail later. The contents of the RAM can be read or written in one machine cycle per word without disturbing any internal registers or status other than the RAM address pointer used for each RAM. The contents of each RAM can be loaded simultaneously into the X and Y inputs of the multiplier.

Registers. The Z89C00 has 12 internal registers and up to an additional eight external registers. The external registers are user definable for peripherals such as A/D or D/A or to DMA or other addressing peripherals. External registers are accessed in one machine cycle the same as internal registers.

FUNCTIONAL DESCRIPTION

General. The Z89C00 is a high-performance Digital Signal Processor with a modified Harvard-type architecture with separate program and data memory. The design has been optimized for processing power and minimizing silicon space.

Instruction Timing. Many instructions are executed in one machine cycle. Long immediate instructions and Jump or Call instructions are executed in two machine cycles. When the program memory is referenced in internal RAM indirect mode, it takes three machine cycles. In addition, one more machine cycle is required if the PC is selected as the destination of a data transfer instruction. This only happens in the case of a register indirect branch instruction.

An Acc + P => Acc; a(i) * b(j) → P calculation and modification of the RAM pointers, is done in one machine cycle. Both operands, a(i) and b(j), can be located in two independent RAM (0 and 1) addresses.

Multiply/Accumulate. The multiplier can perform a 16-bit x 16-bit multiply or multiply accumulate in one machine cycle using the Accumulator and/or both the X and Y inputs. The multiplier produces a 32-bit result, however, only the 24 most significant bits are saved for the next instruction or accumulation. The multiplier provides a flow through operation whenever the X or Y register is updated, an automatic multiply operation is performed and the P register is updated. For operations on very small numbers where the least significant bits are important, the data should first be scaled by eight bits (or the multiplier and multiplicand by four bits each) to avoid truncation errors. Note that all inputs to the multiplier should be fractional two's complement 16-bit binary numbers. This puts them in the range [–1 to 0.9999695], and the result is in 24-bits so that the range is [–1 to 0.9999999]. In addition, if 8000H is loaded into both X and Y registers, the resulting multiplication is considered an illegal operation as an overflow would result. Positive one cannot be represented in fractional notation, and the multiplier will actually yield the result 8000H x 8000H = 8000H (–1 x –1 = –1).

ALU. The 24-bit ALU has two input ports, one of which is connected to the output of the 24-bit Accumulator. The other input is connected to the 24-bit P-Bus, the upper 16 bits of which are connected to the 16-bit D-Bus. A shifter between the P-Bus and the ALU input port can shift the data by three bits right, one bit right, one bit left or no shift.

Hardware Stack. A six-level hardware stack is connected to the D-Bus to hold subroutine return addresses or data. The CALL instruction pushes PC+2 onto the stack. The RET instruction pops the contents of the stack to the PC.

User Inputs. The Z89C00 has two inputs, UI0 and UI1, which may be used by jump and call instructions. The jump or call tests one of these pins and if appropriate, jumps to a new location. Otherwise, the instruction behaves like a NOP. These inputs are also connected to the status register bits S10 and S11 which may be read by the appropriate instruction (Figure 3).

User Outputs. The status register bits S5 and S6 connect through an inverter to UO0 and UO1 pins and may be written to by the appropriate instruction.

Interrupts. The Z89C00 has three positive edge triggered interrupt inputs. An interrupt is acknowledged at the end of any instruction execution. It takes two machine cycles to enter an interrupt instruction sequence. The PC is pushed onto the stack. A RET instruction transfers the contents of the stack to the PC and decrements the stack pointer by one word. The priority of the interrupts is 0 = highest, 2 = lowest.

Registers. The Z89C00 has 12 physical internal registers and up to eight user-defined external registers. The EA2-EA0 determines the address of the external registers. The /EI, /RDYE, and ER/W signals are used to read or write from the external registers.

REGISTERS

There are 12 internal registers which are defined below:

Register	Register Definition
P	Output of Multiplier, 24-Bit, Read Only
X	X Multiplier Input, 16-Bit
Y	Y Multiplier Input, 16-Bit
A	Accumulator, 24-Bit
SR	Status Register, 16-Bit
Pn:b	Six Ram Address Pointers, 8-Bit Each
PC	Program Counter, 16-Bit

The following are virtual registers as physical RAM does not exist on the chip.

EXTn	External registers, 16-bit
BUS	D-Bus
Dn:b	Eight Data Pointers

P holds the result of multiplications and is read-only.

X and **Y** are two 16-bit input registers for the multiplier. These registers can be utilized as temporary registers when the multiplier is not being used. The contents of the P register will change if X or Y is changed.

A is a 24-bit Accumulator. The output of the ALU is sent to this register. When 16-bit data is transferred into this register, it goes into the 16 MSB's and the least significant eight bits are set to zero. Only the upper 16 bits are transferred to the destination register when the Accumulator is selected as a source register in transfer instructions.

Pn:b are the pointer registers for accessing data RAM. (n = 0,1,2 refer to the pointer number) (b = 0,1 refers to RAM bank 0 or 1). They can be directly read from or written to, and can point to locations in data RAM or indirectly to Program Memory.

EXT(n) are external registers (n = 0 to 7). There are eight 16-bit registers here for accessing External data, peripherals, or memory. Note that the actual register RAM does not exist on the chip, but would exist as part of the external device such as an ADC result latch.

BUS is a read-only register which, when accessed, returns the contents of the D-Bus.

Dn:b refer to possible locations in RAM that can be used as a pointer to locations in program memory. The programmer decides which location to choose from two bits in the status register and two bits in the operand. Thus, only the lower 16 possible locations in RAM can be specified. At any one time there are eight usable pointers, four per bank, and the four pointers are in consecutive locations in RAM. For example, if S3/S4 = 01 in the status register, then D0:0/D1:0/D2:0/D3:0 refer to locations 4/5/6/7 in RAM bank 0. Note that when the data pointers are being written to, a number is actually being loaded to Data RAM, so they can be used as a limited method for writing to RAM.

REGISTERS (Continued)

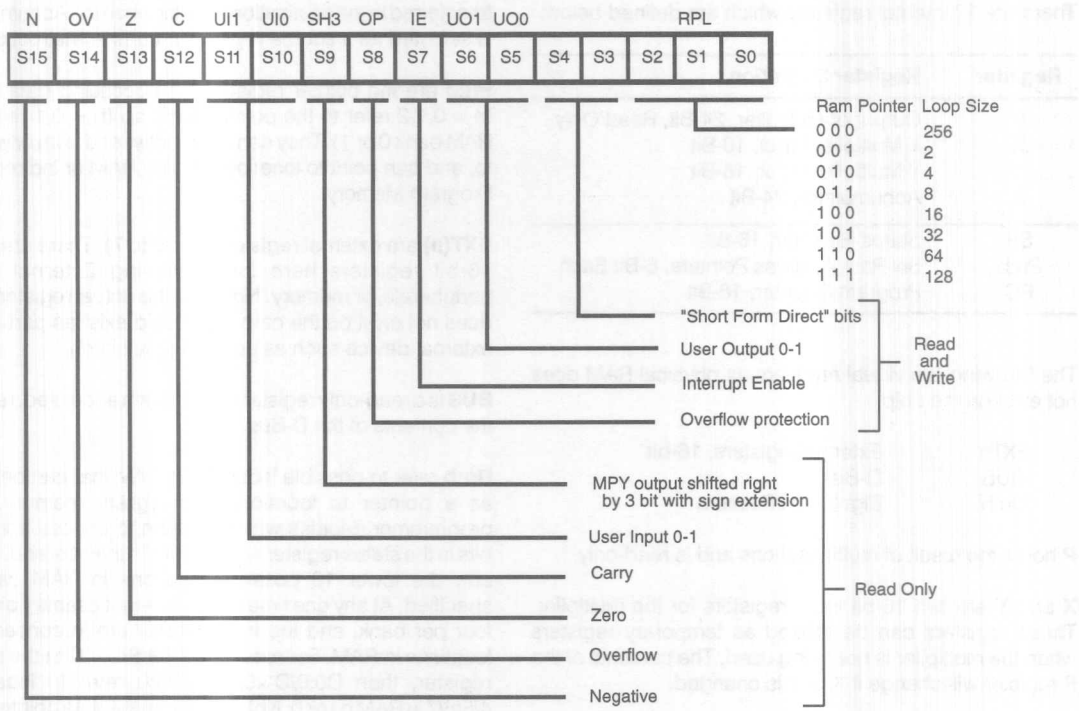


Figure 3. Status Register

ST is the status register (Figure 3) which contains the ALU status and certain control bits as shown in the following table.

Status Register Bit	Function
S15 (N)	ALU Negative
S14 (OV)	ALU Overflow
S13 (Z)	ALU Zero
S12 (L)	Carry
S11 (UI1)	User Input 1
S10 (UI0)	User Input 0
S9 (SH3)	MPY Output Shifted Right by Three Bits
S8 (OP)	Overflow Protection
S7 (IE)	Interrupt Enable
S6 (UO1)	User Output 1
S5 (UO0)	User Output 0
S4-3	"Short Form Direct" Bits
S2-0 (RPL)	RAM Pointer Loop Size

RPL Description

S2	S1	S0	Loop Size
0	0	0	256
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The status register may always be read in its entirety. S15-S10 are set/reset by the hardware and can only be read by software. S9-S0 can be written by software.

S15-S12 are set/reset by the ALU after an operation. S11-10 are set/reset by the user inputs. S6-0 are control bits described elsewhere. S7 enables interrupts. S8, if 0 (reset), allows the hardware to overflow. If S8 is set, the hardware clamps at maximum positive or negative values instead of overflowing. If S9 is set and a multiply instruction is used, the shifter shifts the result three bits right with sign extension.

PC is the Program Counter. When this register is assigned as a destination register, one NOP machine cycle is added automatically to adjust the pipeline timing.

RAM ADDRESSING

The address of the RAM is specified in one of three ways (Figure 4):

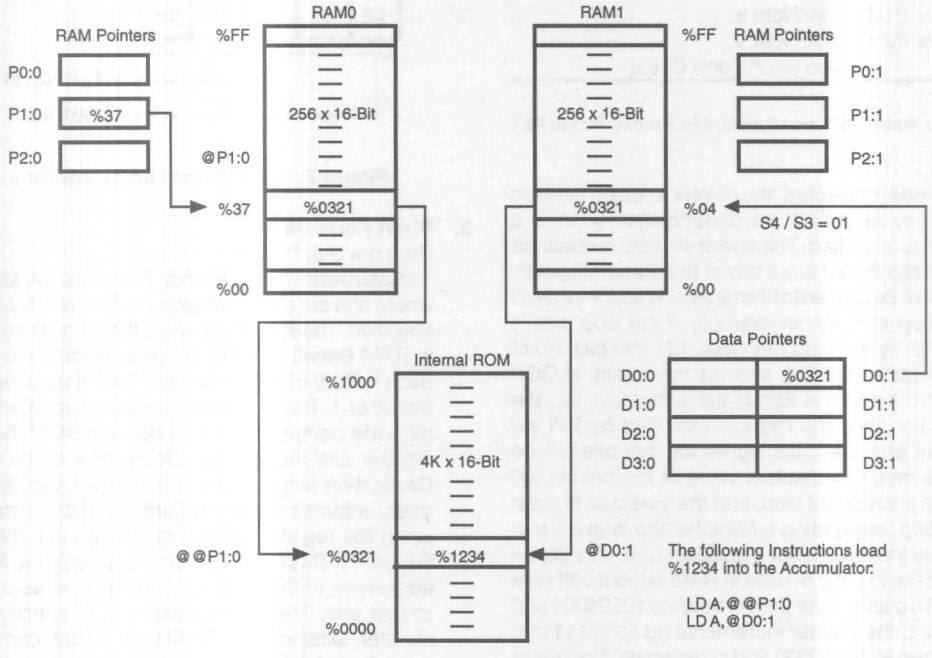


Figure 4. RAM, ROM, and Pointer Architecture

1. **Register Indirect**
   Pn:b n = 0-2, b = 0-1
   The most commonly used method is a register indirect addressing method, where the RAM address is specified by one of the three RAM address pointers (n) for each bank (b). Each source/destination field in Figures 5 and 8 may be used by an indirect instruction to specify a register pointer and its modification after execution of the instruction.

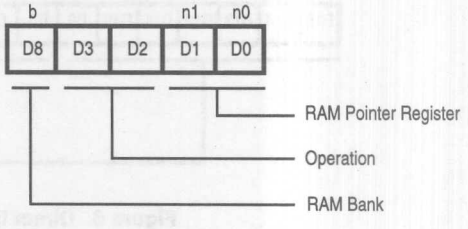


Figure 5. Indirect Register

RAM ADDRESSING (Continued)

The register pointer is specified by the first and second bits in the source/destination field and the modification is specified by the third and fourth bits according to the following table:

D3-D0		Meaning
00xx	NOP	No Operation
01xx	+1	Simple Increment
10xx	–1/LOOP	Decrement Modulo the Loop Count
11xx	+1/LOOP	Increment Modulo the Loop Count
xx00	P0:0 or P0:1	See Note a.
xx01	P1:0 or P1:1	See Note a.
xx10	P2:0 or P2:1	See Note a.
xx11		See Short Form Direct

Note:
a. If Bit 8 is zero, P0:0 to P2:0 are selected; if Bit 8 is one, P0:1 to P2:1 are selected.

When Loop mode is selected, the pointer to which the loop is referring will cycle up or down, depending on whether a –loop or +loop is specified. The size of the loop is obtained from the least significant three bits of the Status Register. The increment or decrement of the register is accomplished modulo the loop size. As an example, if the loop size is specified as 32 by entering the value 101 into bits 2-0 of the Status Register (S2-S0) and an increment +LOOP is specified in the address field of the instruction, i.e., the RPi field is 11xx, then the register specified by RPi will increment, but only the least significant five bits will be affected. This means the actual value of the pointer will cycle round in a length 32 loop, and the lowest or highest value of the loop, depending on whether the loop is up or down, is set by the three most significant bits. This allows repeated access to a set of data in RAM without software intervention. To clarify, if the pointer value is 10101001 and if the loop = 32, the pointer increments up to 10111111, then drops down to 10100000 and starts again. The upper three bits remaining unchanged. Note that the original value of the pointer is not retained.

2. Direct Register

The second method is a direct addressing method. The address of the RAM is directly specified by the address field of the instruction. Because this addressing method consumes nine bits (0-511) of the instruction field, some instructions cannot use this mode (Figure 6).

Figures 8 to 12 show the different register instruction formats along with the two tables below Figure 8.

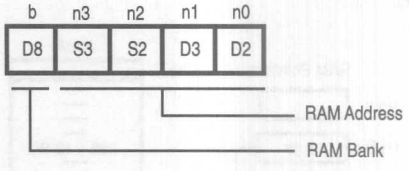


Figure 7. Short Form Direct Address

3. Short Form Direct

Dn:b n = 0-3, b = 0-1

The last method is called Short Form Direct Addressing, where one out of 32 addresses in internal RAM can be specified. The 32 addresses are the 16 lower addresses in RAM Bank0 and the 16 lower addresses in RAM Bank1. Bit 8 of the instruction field determines RAM Bank0 or 1. The 16 addresses are determined by a 4-bit code comprised of bits S3 and S4 of the status register and the third and fourth bits of the Source/Destination field. Because this mode can specify a direct address in a short form, all of the instructions using the register indirect mode can use this mode (Figure 7). This method can access only the lower 16 addresses in the both RAM banks and as such has limited use. The main purpose is to specify a data register, located in the RAM bank, which can then be used to point to a program memory location. This facilitates down-loading look-up tables etc. from program memory to RAM.

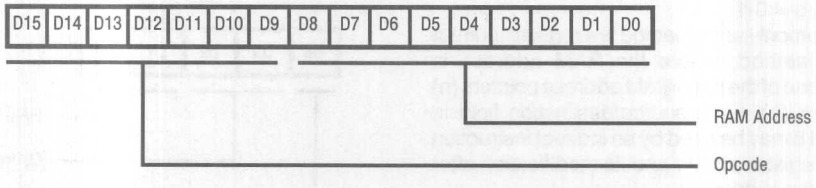
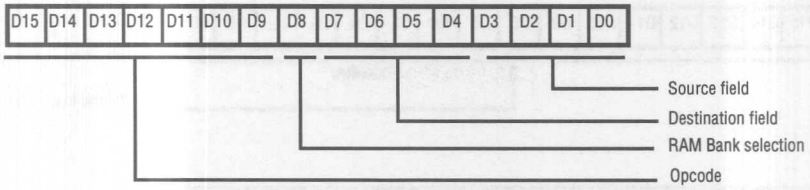


Figure 6. Direct Internal RAM Address Format

INSTRUCTION FORMAT



Note:
Source/Destination fields can specify either register or RAM addresses in RAM pointer indirect mode.

Figure 8. General Instruction Format

A. Registers

Source/Destination	Register
0000	BUS**
0001	X
0010	Y
0011	A
0100	SR
0101	STACK
0110	PC
0111	P**
1000	EXT0
1001	EXT1
1010	EXT2
1011	EXT3
1100	EXT4
1101	EXT5
1110	EXT6
1111	EXT7

B. Register Pointers Field

Source/Destination	Meaning
00xx	NOP
01xx	+1
10xx	–1/LOOP
11xx	+1/LOOP
xx00	P0:0 or P0:1*
xx01	P1:0 or P1:1*
xx10	P2:0 or P2:1*
xx11	Short Form Direct Mode

Notes:
* If RAM Bank bit is 0, then Pn:0 are selected.
If RAM Bank bit is 1, then Pn:1 are selected.
** Read only.

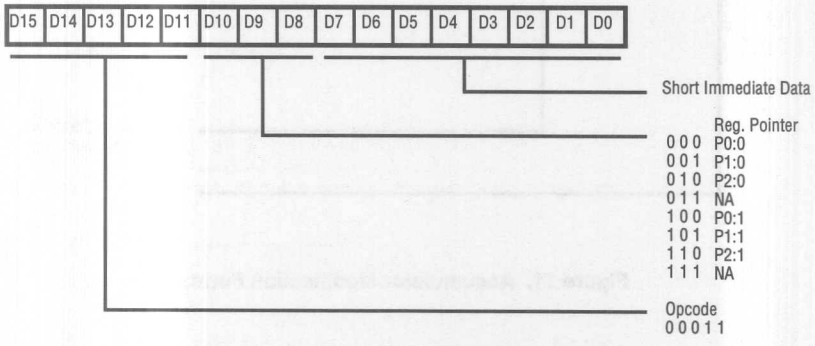


Figure 9. Short Immediate Data Load Format

INSTRUCTION FORMAT (Continued)

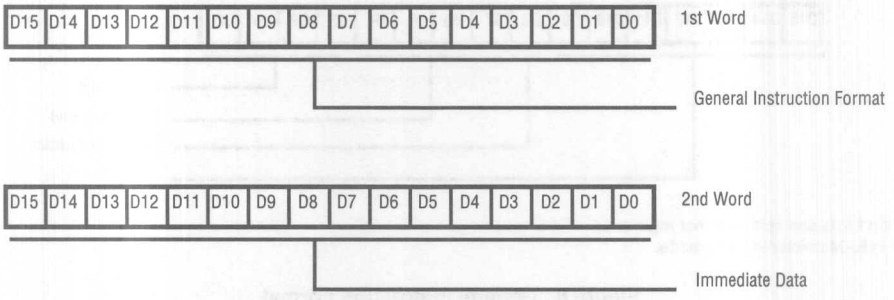


Figure 10. Immediate Data Load Format

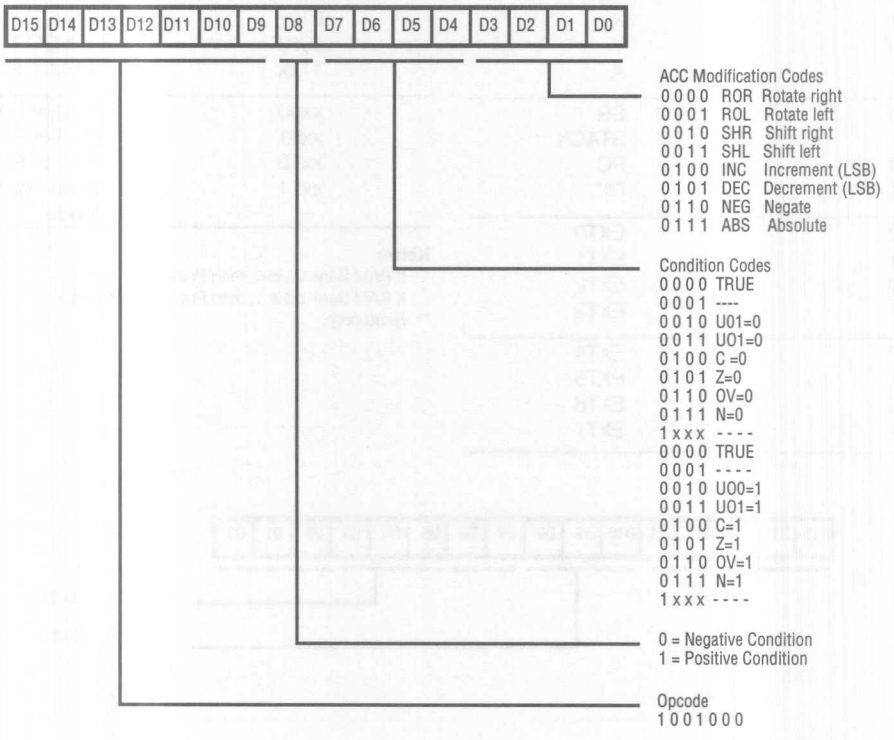


Figure 11. Accumulator Modification Format

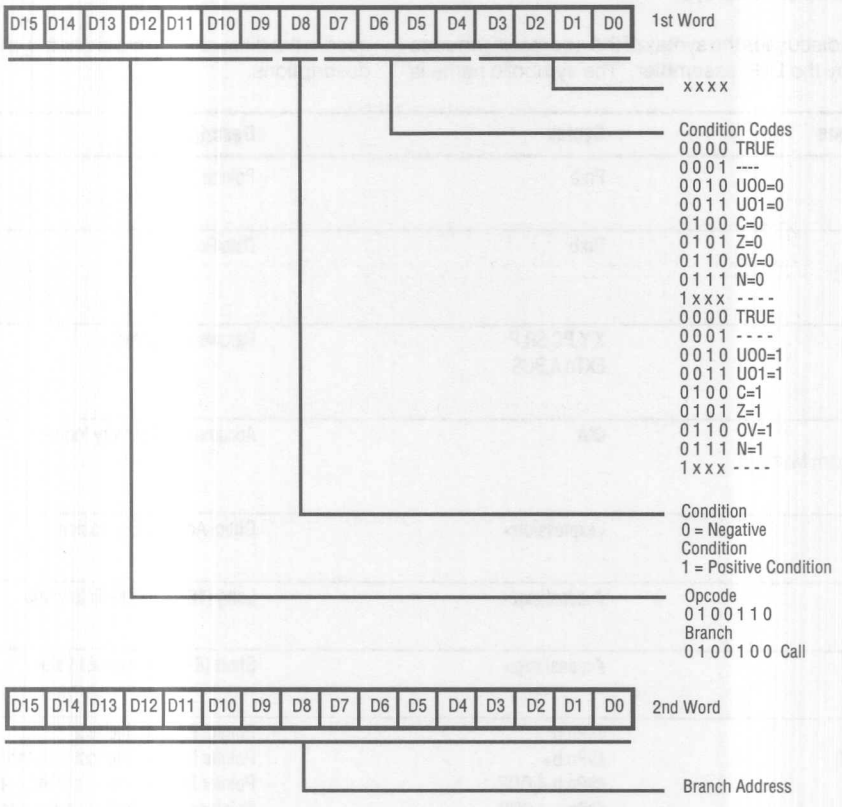


Figure 12. Branching Format

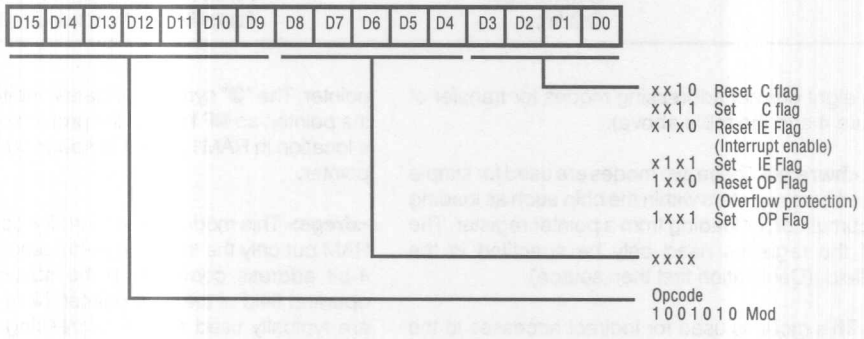


Figure 13. Flag Modification Format

ADDRESSING MODES

This section discusses the syntax of the addressing modes supported by the DSP assembler. The symbolic name is used in the discussion of instruction syntax in the instruction descriptions.

Symbolic Name	Syntax	Description
<pregs>	Pn:b	Pointer Register
<dregs> (Points to RAM)	Dn:b	Data Register
<hwregs>	X,Y,PC,SR,P EXTn,A,BUS	Hardware Registers
<accind> (Points to Program Memory)	@A	Accumulator Memory Indirect
<direct>	<expression>	Direct Address Expression
<limm>	#<const exp>	Long (16-bit) Immediate Value
<simm>	#<const exp>	Short (8-bit) Immediate Value
<regind> (Points to RAM)	@Pn:b @Pn:b+ @Pn:b–LOOP @Pn:b+LOOP	Pointer Register Indirect Pointer Register Indirect with Increment Pointer Register Indirect with Loop Decrement Pointer register Indirect with Loop Increment
<memind> (Points to Program Memory)	@@Pn:b @Dn:b @@Pn:b–LOOP @@Pn:b+LOOP @@Pn:b+	Pointer Register Memory Indirect Data Register Memory Indirect Pointer Register Memory Indirect with Loop Decrement Pointer Register Memory Indirect with Loop Increment Pointer Register Memory Indirect with Increment

There are eight distinct addressing modes for transfer of data (Figure 4 and the table above).

<pregs>, <hwregs> These two modes are used for simple loads to and from registers within the chip such as loading to the Accumulator, or loading from a pointer register. The names of the registers need only be specified in the operand field. (Destination first then source)

<regind> This mode is used for indirect accesses to the data RAM. The address of the RAM location is stored in the pointer. The "@" symbol indicates "indirect" and precedes the pointer, so @P1:1 tells the processor to read or write to a location in RAM1, which is specified by the value in the pointer.

<dregs> This mode is also used for accesses to the data RAM but only the lower 16 addresses in either bank. The 4-bit address comes from the status register and the operand field of the data pointer. Note that data registers are typically used not for addressing RAM, but loading data from program memory space.

<memind> This mode is used for indirect, indirect accesses to the program memory. The address of the memory is located in a RAM location, which is specified by the value in a pointer. So @@P1:1 tells the processor to read (write is not possible) from a location in memory, which is specified by a value in RAM, and the location of the RAM is in turn specified by the value in the pointer. Note that the data pointer can also be used for a memory access in this manner, but only one "@" precedes the pointer. In both cases the memory address stored in RAM is incremented by one each time the addressing mode is used to allow easy transfer of sequential data from program memory.

<accind> Similar to the previous mode, the address for the program memory read is stored in the Accumulator. @A in the second operand field loads the number in memory specified by the address in A.

<direct> The direct mode allows read or write to data RAM from the Accumulator by specifying the absolute address of the RAM in the operand of the instruction. A number between 0 and 255 indicates a location in RAM0, and a number between 256 and 511 indicates a location in RAM1.

<limm> This indicates a long immediate load. A 16-bit word can be copied directly from the operand into the specified register or memory.

<simm> This can only be used for immediate transfer of 8-bit data in the operand to the specified RAM pointer.

1

CONDITION CODES

The following table defines the condition codes supported by the DSP assembler. If the instruction description refers to the <cc> (condition code) symbol in one of its addressing modes, the instruction will only execute if the condition is true.

Name	Description
C	Carry
EQ	Equal (same as Z)
F	False
IE	Interrupts Enabled
MI	Minus
NC	No Carry
NE	Not Equal (same as NZ)
NIE	Not Interrupts Enabled
NOV	Not Overflow
NU0	Not User Zero

Name	Description
NU1	Not User One
NZ	Not zero
OV	Overflow
PL	Plus (Positive)
U0	User Zero
U1	User One
UGE	Unsigned Greater Than or Equal (Same as NC)
ULT	Unsigned Less Than (Same as C)
Z	Zero

INSTRUCTION DESCRIPTIONS

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
ABS	Absolute Value	ABS[<cc>,]<src>	<cc>,A	1	1	ABS NC,A
			A	1	1	ABS A
ADD	Addition	ADD<dest>,<src>	A,<pregs>	1	1	ADD A,P0:0
			A,<dregs>	1	1	ADD A,D0:0
			A,<limm>	2	2	ADD A,#%1234
			A,<memind>	1	3	ADD A,@@P0:0
			A,<direct>	1	1	ADD A,%F2
			A,<regind>	1	1	ADD A,@P1:1
			A,<hwregs>	1	1	ADD A,X
AND	Bitwise AND	AND<dest>,<src>	A,<pregs>	1	1	AND A,P2:0
			A,<dregs>	1	1	AND A,D0:1
			A,<limm>	2	2	AND A,#%1234
			A,<memind>	1	3	AND A,@@P1:0
			A,<direct>	1	1	AND A,%2C
			A,<regind>	1	1	AND A,@P1:2+LOOP
			A,<hwregs>	1	1	AND A,EXT3
CALL	Subroutine call	CALL [<cc>,]<address>	<cc>,<direct>	2	2	CALL Z,sub2
			<direct>	2	2	CALL sub1
CCF	Clear carry flag	CCF	None	1	1	CCF
CIEF	Clear Carry Flag	CIEF	None	1	1	CIEF
COPF	Clear OP flag	COPF	None	1	1	COPF
CP	Comparison	CP<src1>,<src2>	A,<pregs>	1	1	CP A,P0:0
			A,<dregs>	1	1	CP A,D3:1
			A,<memind>	1	3	CP A,@@P0:1
			A,<direct>	1	1	CP A,%FF
			A,<regind>	1	1	CP A,@P2:1+
			A,<hwregs>	1	1	CP A,STACK
			A<limm>	2	2	CP A,#%FFCF
DEC	Decrement	DEC [<cc>,]<dest>	<cc>A,	1	1	DEC NZ,A
			A	1	1	DEC A
INC	Increment	INC [<cc>,] <dest>	<cc>,A	1	1	INC PL,A
			A	1	1	INC A
JP	Jump	JP [<cc>,]<address>	<cc>,<direct>	2	2	JP NIE,Label
			<direct>	2	2	JP Label

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
LD	Load destination with source	LD<dest>,<src>	A,<hwregs>	1	1	LD A,X
			A,<dregs>	1	1	LD A,D0:0
			A,<pregs>	1	1	LD A,P0:1
			A,<regind>	1	1	LD A,@P1:1
			A,<memind>	1	3	LD A,@D0:0
			A,<direct>	1	1	LD A,124
			<direct>,A	1	1	LD 124,A
			<dregs>,<hwregs>	1	1	LD D0:0,EXT7
			<pregs>,<simm>	1	1	LD P1:1,#%FA
			<pregs>,<hwregs>	1	1	LD P1:1,EXT1
			<regind>,<limm>	1	1	LD@P1:1,#1234
			<regind>,<hwregs>	1	1	LD @P1:1+,X
			<hwregs>,<pregs>	1	1	LD Y,P0:0
			<hwregs>,<dregs>	1	1	LD SR,D0:0
			<hwregs>,<limm>	2	2	LD PC,#%1234
			<hwregs>,<accind>	1	3	LD X,@A
			<hwregs>,<memind>	1	3	LD Y,@D0:0
			<hwregs>,<regind>	1	1	LD A,@P0:0–LOOP
			<hwregs>,<hwregs>	1	1	LD X,EXT6
			Note: If X or Y register is the destination, an automatic multiply operation is performed.			
			Note: The P register is Read Only and cannot be destination.			
			Note: LD EXT_N, EXT_N is not allowed.			
			Note: LD A, @A is not allowed.			
MLD	Multiply	MLD<src1>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MLD A,@P0:0+LOOP
			<hwregs>,<regind>,<bank switch>	1	1	MLD A,@P1:0,OFF
			<regind>,<regind>	1	1	MLD @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MLD @P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind must be a bank 0 register.			
			Note: <hwregs> for src1 cannot be X.			
			Note: For the operands <hwregs>, <regind> the <band switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			
MPYA	Multiply and add	MPYA <src1>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYA A,@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYA A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYA @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYA@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register.			
			Note: <hwregs> for src1 cannot be X or A.			
			Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			

INSTRUCTION DESCRIPTIONS (Continued)

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
MPYS	Multiply and subtract	MPYS<src1>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYS A,@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYS A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYS @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYS@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register.			
			Note: <hwregs> for src1 cannot be X or A.			
			Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, <regind> the <bank switch> defaults to ON.			
NEG	Negate	NEG <cc>,A	<cc>, A	1	1	NEG MI,A
			A	1	1	NEG A
NOP	No operation	NOP	None	1	1	NOP
OR	Bitwise OR	OR <dest>,<src>	A, <pregs>	1	1	OR A,P0:1
			A, <dregs>	1	1	OR A, D0:1
			A, <limm>	2	2	OR A,#%2C21
			A, <memind>	1	3	OR A,@@P2:1+
			A, <direct>	1	1	OR A, %2C
			A, <regind>	1	1	OR A,@P1:0–LOOP
			A, <hwregs>	1	1	OR A,EXT6
POP	Pop value from stack	POP <dest>	<pregs>	1	1	POP P0:0
			<dregs>	1	1	POP D0:1
			<regind>	1	1	POP @P0:0
			<hwregs>	1	1	POP A
PUSH	Push value onto stack	PUSH <src>	<pregs>	1	1	PUSH P0:0
			<dregs>	1	1	PUSH D0:1
			<regind>	1	1	PUSH @P0:0
			<hwregs>	1	1	PUSH BUS
			<limm>	2	2	PUSH #12345
			<accind>	1	3	PUSH @A
			<memind>	1	3	PUSH @@P0:0
RET	Return from subroutine	RET	None	1	2	RET
RL	Rotate Left	RL <cc>,A	<cc>,A	1	1	RL NZ,A
			A	1	1	RL A
RR	Rotate Right	RR <cc>,A	<cc>,A	1	1	RR C,A
			A	1	1	RR A

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
SCF	Set C flag	SCF	None	1	1	SCF
SIEF	Set IE flag	SIEF	None	1	1	SIEF
SLL	Shift left logical	SLL	[<cc>,]A	1	1	SLL NZ,A
			A	1	1	SLL A
SOPF	Set OP flag	SOPF	None	1	1	SOPF
SRA	Shift right arithmetic	SRA<cc>,A	<cc>,A	1	1	SRA NZ,A
			A	1	1	SRA A
SUB	Subtract	SUB<dest>,<src>	A,<pregs>	1	1	SUB A,P1:1
			A,<dregs>	1	1	SUB A,D0:1
			A,<limm>	2	2	SUB A,#%2C2C
			A, <memind>	1	3	SUB A,@D0:1
			A, <direct>	1	1	SUB A,%15
			A, <regind>	1	1	SUB A,@P2:0-LOOP
			A, <hwregs>	1	1	SUB A,STACK
XOR	Bitwise exclusive OR	XOR <dest>,<src>	A, <pregs>	1	1	XOR A,P2:0
			A, <dregs>	1	1	XOR A,D0:1
			A, <limm>	2	2	XOR A,#13933
			A, <memind>	1	3	XOR A,@@P2:1+
			A, <direct>	1	1	XOR A,%2F
			A, <regind>	1	1	XOR A,@P2:0
			A, <hwregs>	1	1	XOR A,BUS

Bank Switch Enumerations. The third (optional) operand of the MLD, MPYA and MPYS instructions represents whether a bank switch is set on or off. To more clearly represent this, two keywords are used (ON and OFF) which state the direction of the switch. These keywords are referred to in the instruction descriptions through the <bank switch> symbol.

ABSOLUTE MAXIMUM RATINGS

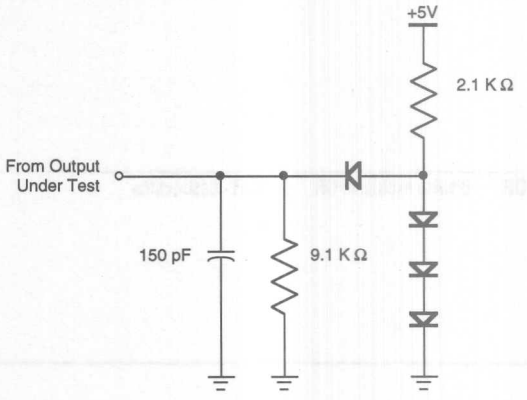
Symbol	Description	Min	Max	Units
V_{CC}	Supply Voltage(*)	–0.5	7.0	V
T_{STG}	Storage Temp.	–65°	+150°	C
T_A	Oper. Ambient Temp.	†		C

Notes:
* Voltages on all pins with respect to ground.
† See Ordering Information

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended period may affect device reliability.

STANDARD TEST CONDITIONS

The characteristics listed below apply for standard test conditions as noted. All voltages are referenced to ground. Positive current flows into the referenced pin (Test Load Diagram).



Test Load Diagram

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5V ± 5%, T_A = 0°C to +70°C unless otherwise specified)

Symbol	Parameter	Condition	Min.	Max.	Units
I_{CC}	Supply Current	V_{CC} = 5.25V fclock = 10 MHz		60	mA
I_{CC1}	Halt Mode	V_{CC} = 5.25V fclock = 0 MHz (stopped)	1	5	mA
V_{IH}	Input High Level		0.9 V_{CC}		V
V_{IL}	Input Low Level			0.1 V_{CC}	V
I_{IL}	Input Leakage			1	µA
V_{OH}	Output High Voltage	I_{OH} = –100 µA	V_{CC} –0.2		V
V_{OL}	Output Low Voltage	I_{OL} = 0.5 mA		0.5	V
I_{FL}	Output Floating Leakage Current			5	µA

AC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5V ± 5%, T_A = 0°C to +70°C unless otherwise specified)

No.	Symbol	Parameter	Min.	Max.	Units
1	TCY	Clock Cycle Time	100	1000	ns
2	PWW	Clock Pulse Width	45		ns
3	Tr	Clock Rise Time	2	4	ns
4	Tf	Clock Fall Time	2	4	ns
5	TEAD	EA,ER//W Delay from CK	9	33	ns
6	TXVD	EXT Data Output Valid from CLK	5	27	ns
7	TXWH	EXT Data Output Hold from CLK	6	22	ns
8	TXRS	EXT Data Input Setup Time	15		ns
9	TXRH	EXT Data Input Hold from CLK	5	15	ns
10	TIEDR	/EI Delay Time from Rising CLK Edge	3	15	ns
11	TIEDF	/EI Delay Time from Falling CLK Edge	0	23	ns
12	TINS	Interrupt Setup Time	5		ns
13	TINL	Interrupt Hold Time	15		ns
14	TPAD	PA Delay from CLK	5	22	ns
15	TPDS	PD Input Setup Time	20		ns
16	TPDH	PD Input Hold Time	20	28	ns
17	TCTLS	Halt Setup Time	5		ns
18	TCTLH	Halt Hold Time	20		ns
19	RDYS	Ready Setup Time	10		ns
20	RDYH	Ready Hold Time	7		ns

AC TIMING DIAGRAM

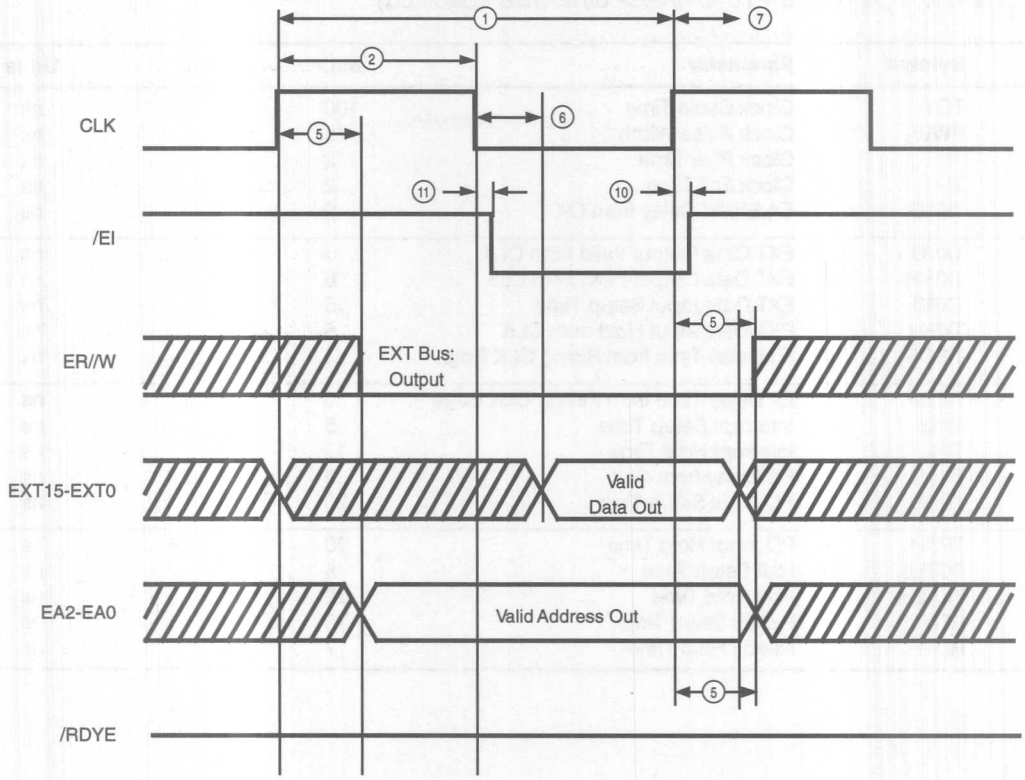


Figure 14a. WRITE To External Device Timing

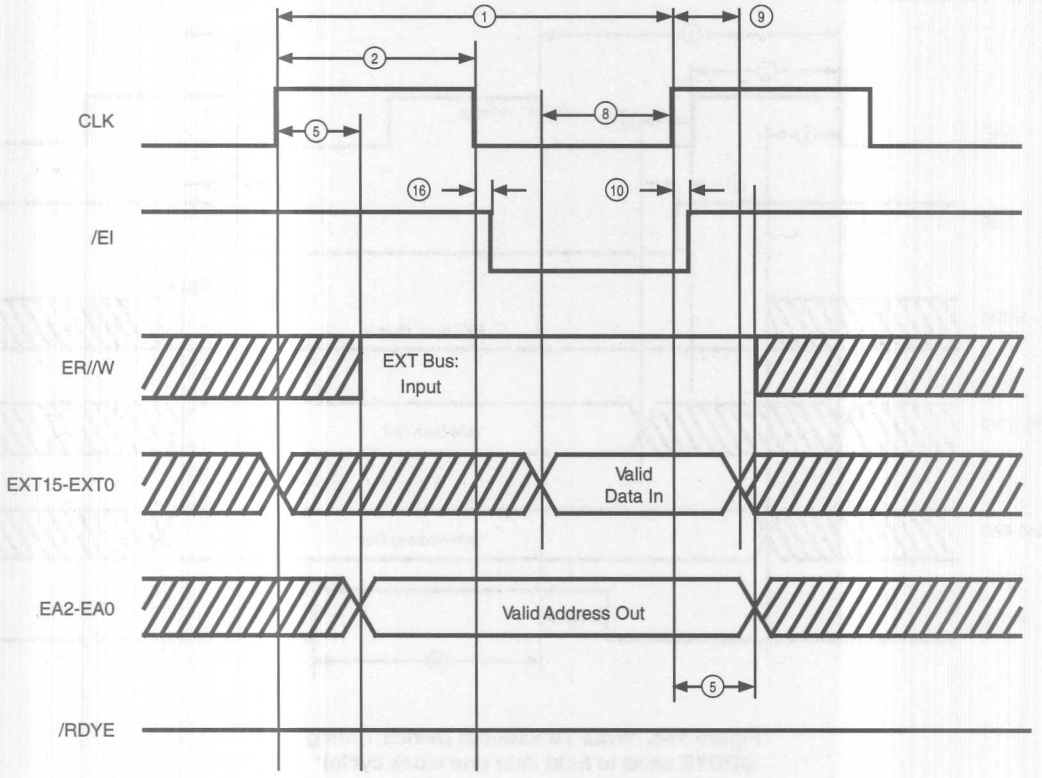


Figure 14b. READ From External Device Timing

AC TIMING DIAGRAM

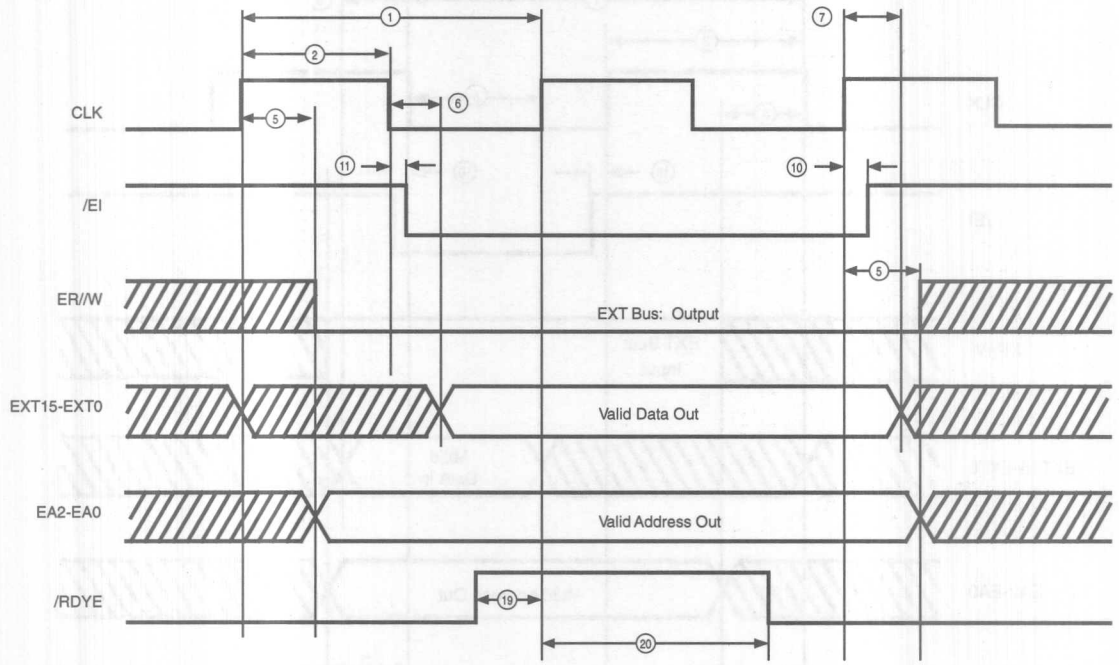


Figure 14c. Write To External Device Timing (/RDYE used to hold data one clock cycle)*

Note: * /RDYE is checked during rising edge of clock.

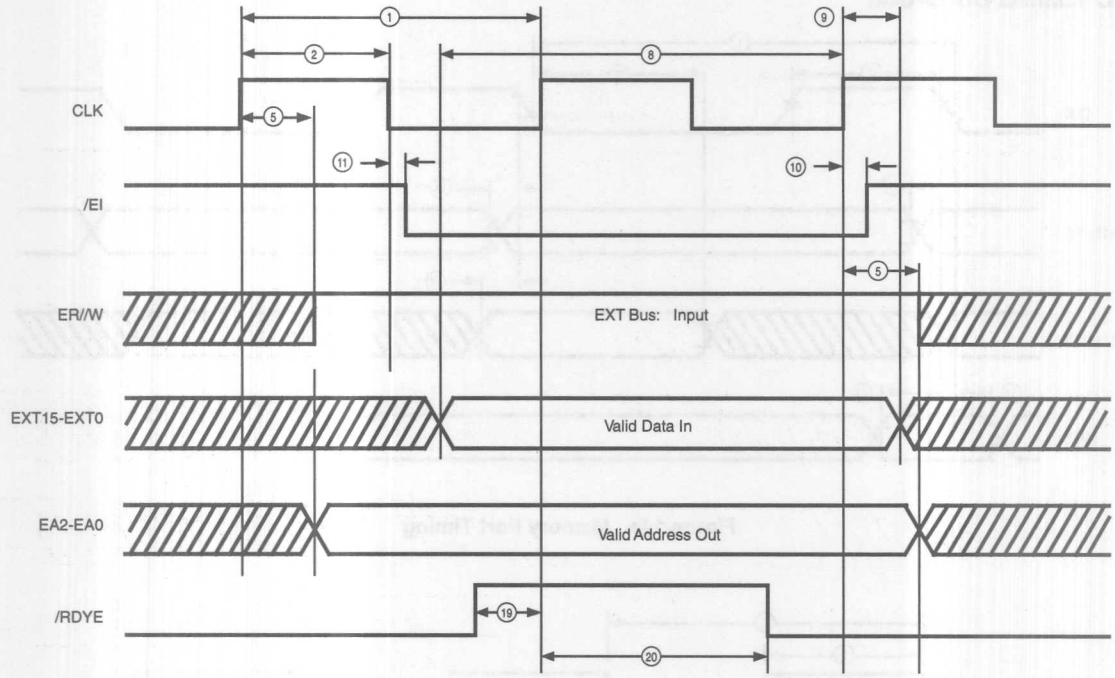


Figure 14d. Read From External Device Timing (/RDYE used to hold data one clock cycle)*

Note: * /RDYE is checked during rising edge of clock.

AC TIMING DIAGRAM

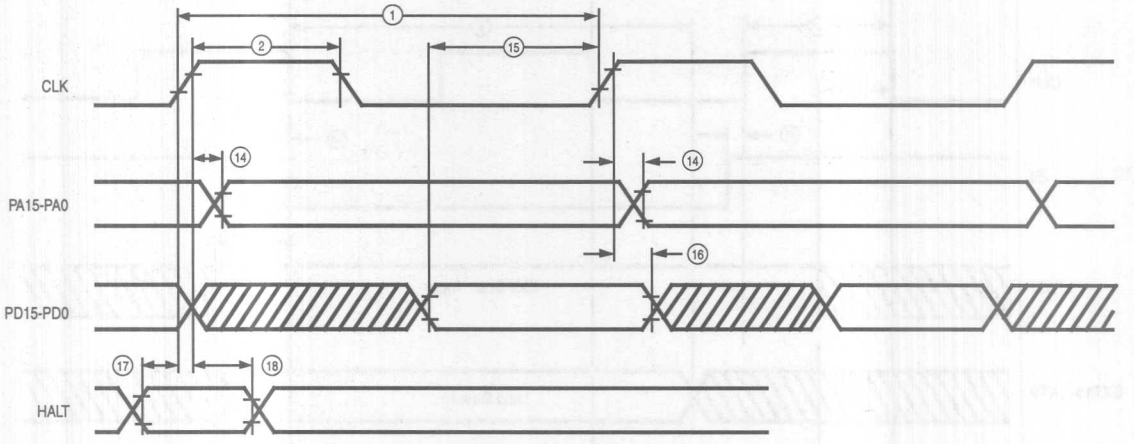


Figure 14e. Memory Port Timing

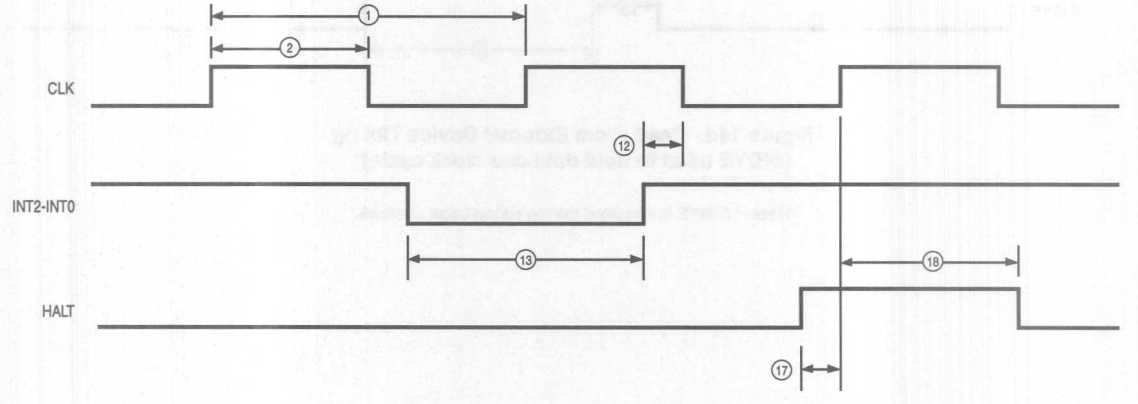
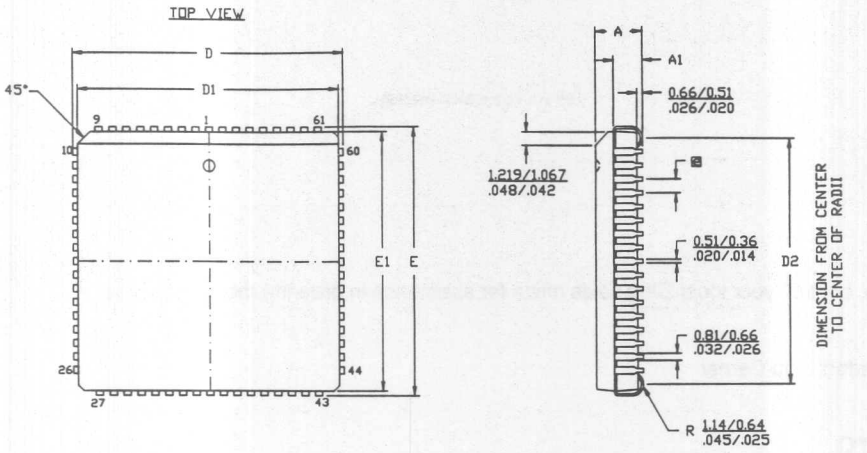


Figure 14f. Interrupt and HALT Timing

PACKAGE INFORMATION



NOTES:

1. CONTROLLING DIMENSIONS : INCH
2. LEADS ARE COPLANAR WITHIN .004 IN.
3. DIMENSION : MM / INCH

SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.32	4.57	.170	.180
A1	2.67	2.92	.105	.115
D/E	25.02	25.40	.985	1.000
D1/E1	24.13	24.33	.950	.958
D2	22.86	23.62	.900	.930
e	1.27 TYP		.050 TYP	

68-Pin PLCC Package Diagram

ORDERING INFORMATION

Z89C00

10 MHz
68-pin PLCC
Z89C0010VSC

15 MHz
68-pin PLCC
Z89C0015VSC

For fast results, contact your local Zilog sales office for assistance in ordering the part desired.

Package
V = Plastic Leaded Chip Carrier

Temperature
S = 0°C to +70°C

Speeds
10 = 10 MHz
15 = 15 MHz

Environmental
C = Plastic Standard

Example:

Z 89C00 10 V S C is a Z89C00, 10 MHz, PLCC, 0°C to +70°C, Plastic Standard Flow

- C — Environmental Flow
- S — Temperature
- V — Package
- 10 — Speed
- 89C00 — Product Number
- Z — Zilog Prefix

Z89C00 DSP
UNDERSTANDING Q15 TWO'S COMPLEMENT FRACTIONAL MULTIPLICATION

DSP multiply/accumulate instructions are computed in a single machine cycle with the Zilog Z89C00. This Application Note describes the two's complement fractional multiplication process.

INTRODUCTION

DSP (Digital Signal Processor) multiplication is broken down into its elements in this Application Note. All the DSP requires is that the values to be multiplied are entered and the result will be given in one cycle.

The Zilog Z89C00 uses fixed point, two's complement fractional representation. The left-most bit (MSB) has a sign or weight of –1 and all the remaining bits have positive fractional weights (Q15). Values to be multiplied are written into the X (16-bit) and Y (16-bit) registers and the result (32-bit) is read in the P register. Note that only the 24 most significant bits are saved for the next instruction or accumulation.

The following working examples can be used with either a scientific calculator, a Lotus® 1-2-3® spreadsheet, or a Z89C00 Zilog DSP GUI emulator or PLC simulator/emulator interface.

SCIENTIFIC CALCULATOR

The calculator should, in addition to all the regular functions, have hexadecimal to decimal (HEX to DEC, DEC to HEX) and exclusive OR (XOR). Before beginning the multiplication, a couple of quick conversions from two's complement fractional notation numbers to decimal:

Converting Positive Numbers:

6FFF	HEX → 28671	DEC/32768	= 0.8749	DEC	
5ABC	HEX → 23228	DEC/32768	= 0.7088	DEC	
28AD	HEX → 10413	DEC/32768	= 0.3177	DEC	
0.8749	DEC • 32768 = 28671	DEC → 6FFF	HEX		
0.7088	DEC • 32768 = 23228	DEC → 5ABC	HEX		
0.3177	DEC • 32768 = 10413	DEC → 28AD	HEX		

Converting Negative Numbers:

HEX	Two's Complement	DEC Result
8000	XOR with FFFF + 1 = 8000 → DEC 32768/32768 • –1 =	–1.0000
8008	XOR with FFFF + 1 = 7FF8 → DEC 32760/32768 • –1 =	–0.9997
9ABC	XOR with FFFF + 1 = 6544 → DEC 25924/32768 • –1 =	–0.7911

DEC	Two's Complement	HEX Result
–1.0000 • –1 = 1.0000 • 32768 to HEX → 8000	XOR with FFFF + 1 =	8000
–0.9997 • –1 = 0.9997 • 32768 to HEX → 7FF6	XOR with FFFF + 1 =	8008
–0.7911 • –1 = 0.7911 • 32768 to HEX → 6542	XOR with FFFF + 1 =	9ABC

8000H is 32768 (215 plus a sign bit) decimal. Everything that has a range of ±32768 to a range of –1 (the sign bit has a weight of –1) to +0.9999695 is being rescaled, hence, the division and multiplication by 32768. Note: Recorder can be taught to perform this function by using the windows calculator.

One line of code is required before implementing the multiplication examples:

Instruction	Operation
MLD @P0:1,@P0:0	@P0:1 • @P0:0

This instruction multiplies two RAM operands. The format requires that RAM Bank 1 be referenced first.

Example 1: (+ve) • (+ve)

Multiplying a positive two's complement fractional notation number with a positive two's complement fractional notation number:

((+ve) • (+ve)) •2)

a. Multiply
b. Shift left 1 bit (Multiply by 2)
c. Use only top 4 bytes

6FFF • 6FFF

```
      6FFF   (Place in X register)
    × 6FFF   (Place in Y register)
  30FF2001
    ×    2   (Multiply •2 equivalent to shift left 1 bit)
  61FE4002

    61FE40   (Save only top 6 bytes. This is what appears in the P register)
      61FE   (Use only top four bytes)
```

In summary,

6FFF HEX (0.8749 DEC) • 6FFF HEX (0.8749 DEC) = 61FE HEX (0.7655 DEC)

Example 2: (–ve) • (–ve)

Multiplying a negative two's complement fractional notation number with a negative two's complement fractional notation number:

((XOR(–ve,FFFF)+1) • (XOR(–ve,FFFF)+1)) •2

a. Take Two's Complement of both –ve negative numbers (XOR with FFFF then add 1)
b. Multiply results
c. Shift left 1 bit (Multiply by 2)
d. Use only top 4 bytes

9DAB • AEAF

```
      6255   (Two's Complement of 9DAB, i.e., XOR(9DAB,FFFF) +1)
    × 5151   (Two's Complement of AEAF, i.e., XOR(AEAF,FFFF) +1)
  1F3C01E5
    ×    2   (Multiply •2 equivalent to shift left 1 bit)
  3E7803CA

    3E7803   (Save only top 6 bytes. This is what appears in the P register)
      3E78   (Use only top four bytes)
```

In summary,

9DAB HEX (–0.7682 DEC) • AEAF HEX (–0.6352 DEC) = 3E78 HEX (0.4880 DEC)

Example 3: (–ve) • (+ve)

Multiplying a negative two's complement fractional notation number with a positive two's complement fractional notation number:

(XOR((((XOR(–ve,FFFF)+1) • (+ve)) •2),FFFF)) +1

a. Take Two's Complement of –ve (XOR with FFFF then add 1)
b. Multiply result with +ve
c. Shift left 1 bit (Multiply by 2)
d. Take Two's Complement –ve (XOR with FFFF then add 1)
e. Use only top 4 bytes

```
9DAB • 6FFF
    6255   (Two's Complement of 9DAB, i.e., XOR(9DAB,FFFF) +1)
×   6FFF
2B04CDAB
×      2   (Multiply •2 equivalent to shift left 1 bit)
56099B56

A9F664AA   (Two's Complement)
A9F664     (Save only top 6 bytes. This is what appears in the P register)
A9F6       (Use only top four bytes)
```

In summary,

9DAB HEX (–0.7682 DEC) • 6FFF HEX (0.8749 DEC) = A9F6 HEX (–0.6721)

Be especially careful to note the truncations that take place which are normally implicit in this type of mathematics. Although the multiplier produces a 32-bit result, only the 24 most significant bits are saved. For example, if this result is saved temporarily in a RAM BANK, then only the 16 most significant bits are saved. If greater precision (using very small numbers) is required, then the data should be scaled before doing the multiplication.

NOTE: The DSP is designed for two's complement fractional multiplies and is useless for scalar multiplication.

LOTUS

LOTUS spreadsheet software can be used to make conversions and do two's complement fractional multiplication. Convert from HEX to decimal and multiply both decimal numbers together and reconvert the result to HEX. When converting from decimal to HEX, use HEX $16^0 = 1$, $16^1 = 16$, $16^2 = 256$, and $16^3 = 4096$. The equivalent decimal value is checked for sign, and, by dividing by 16^3, 16^2, 16^1, and 16^0, is converted to HEX. This version was tested and written using a UNIX™ version of LOTUS but it should work on most versions of LOTUS. LOTUS does not have a hexadecimal format. However, conversions can still be done as shown in the following example:

Q15

NOTE: Enter hex value in "HEX to" as a character string, i.e., '6ff9.

DEC to	HEX	•32768	4096	256	16	1				
–0.67217	A9F6	43510.39	10.62265	9.962474	15.39958	6.393219	A	9	F	6
HEX to	DEC									
9dab	–0.76822	9	13	10	11	40363	9	d	a	b
6fff	0.874969	6	15	15	15	28671	6	f	f	f
	–0.67217									

In the spreadsheet, enter the two numbers to be multiplied 9dab and 6fff as above. These numbers are converted to decimal, multiplied together, and then reconverted to HEX.

In summary,

9DAB HEX (–0.7682 DEC) • 6FFF HEX (0.8749 DEC) = A9F6 HEX (–0.6721)

The equations used for the spreadsheet are listed below row by row, but must be entered in the appropriate cells as indicated below:

```
A:A1:  [W9]  'Q15
A:C1:  [W9]  'NOTE: Enter hex value in "HEX to" as character string, i.e., '6ff9
A:A3:  [W9]  'DEC to
A:C3:  [W9]  'HEX
A:D3:  [W9]  '*32768
A:E3:  [W9]  4096
A:F3:  [W9]  256
A:G3:  [W9]  16
A:H3:  [W9]  1
A:A5:  [W9]  +C11
A:C5:  [W9]  +I5&J5&K5&L5
A:D5:  [W9]  (@IF((A5*32768)>1,(A5*32768),(2^16+(A5*32768))))
A:E5:  [W9]  (D5/$E$3)
A:F5:  [W9]  ((E5-(@INT(E5)))*$E$3/$F$3)
A:G5:  [W9]  ((F5-(@INT(F5)))*$F$3)/$G$3
A:H5:  [W9]  (G5-@INT(G5))*$G$3
A:I5:  [W2]  @CHOOSE((@INT(E5)),"0","1","2","3","4","5","6","7","8","9","A","B","C","D","E","F","0")
A:J5:  [W2]  @CHOOSE((@INT(F5)),"0","1","2","3","4","5","6","7","8","9","A","B","C","D","E","F","0")
A:K5:  [W2]  @CHOOSE((@INT(G5)),"0","1","2","3","4","5","6","7","8","9","A","B","C","D","E","F","0")
A:L5:  [W2]  @CHOOSE((@INT(H5)),"0","1","2","3","4","5","6","7","8","9","A","B","C","D","E","F","0")
A:A7:  [W9]  'HEX to
A:C7:  [W9]  'DEC
A:A9:  [W9]  '9dab
A:C9:  [W9]  (@IF((H9/32768)<1,(H9/32768),((-2^16+H9)/32768)))
A:D9:  [W9]  @IF(@ISSTRING(I9),@CODE(I9)-87,I9)
A:E9:  [W9]  @IF(@ISSTRING(J9),@CODE(J9)-87,J9)
A:F9:  [W9]  @IF(@ISSTRING(K9),@CODE(K9)-87,K9)
A:G9:  [W9]  @IF(@ISSTRING(L9),@CODE(L9)-87,L9)
A:H9:  [W9]  +D9*$E$3+E9*$F$3+F9*$G$3+G9
A:I9:  [W2]  @IF(@CODE(@MID($A9,0,1))<58,@VALUE(@MID($A9,0,1)),@MID($A9,0,1))
A:J9:  [W2]  @IF(@CODE(@MID($A9,1,1))<58,@VALUE(@MID($A9,1,1)),@MID($A9,1,1))
A:K9:  [W2]  @IF(@CODE(@MID($A9,2,1))<58,@VALUE(@MID($A9,2,1)),@MID($A9,2,1))
A:L9:  [W2]  @IF(@CODE(@MID($A9,3,1))<58,@VALUE(@MID($A9,3,1)),@MID($A9,3,1))
A:A10: [W9]  '6fff
A:C10: [W9]  (@IF((H10/32768)<1,(H10/32768),((-2^16+H10)/32768)))
A:D10: [W9]  @IF(@ISSTRING(I10),@CODE(I10)-87,I10)
A:E10: [W9]  @IF(@ISSTRING(J10),@CODE(J10)-87,J10)
A:F10: [W9]  @IF(@ISSTRING(K10),@CODE(K10)-87,K10)
A:G10: [W9]  @IF(@ISSTRING(L10),@CODE(L10)-87,L10)
A:H10: [W9]  +D10*$E$3+E10*$F$3+F10*$G$3+G10
A:I10: [W2]  @IF(@CODE(@MID($A10,0,1))<58,@VALUE(@MID($A10,0,1)),@MID($A10,0,1))
A:J10: [W2]  @IF(@CODE(@MID($A10,1,1))<58,@VALUE(@MID($A10,1,1)),@MID($A10,1,1))
A:K10: [W2]  @IF(@CODE(@MID($A10,2,1))<58,@VALUE(@MID($A10,2,1)),@MID($A10,2,1))
A:L10: [W2]  @IF(@CODE(@MID($A10,3,1))<58,@VALUE(@MID($A10,3,1)),@MID($A10,3,1))
A:C11: [W9]  +C9*C10
```

Lotus and 1-2-3 are registered trademarks of Lotus Development Corp.
UNIX is a trademark of Bell Laboratories.

Z89120
Z89920 (ROMLESS)
16-BIT MIXED SIGNAL PROCESSOR

FEATURES

- Z8® Microcontroller with 47 I/O Lines
- 24 Kbytes of Z8 Program ROM (Z89120)
- 256 Bytes On-Chip Z8 RAM
- Watch-Dog Timer and Power-On Reset
- Low Power STOP Mode
- On-Chip Oscillator which Accepts a Crystal or External Clock Drive
- Two 8-Bit Z8 Counter Timers with 6-Bit Prescaler
- Global Power-Down Mode
- Low Power Consumption - 200 mW (Typical)
- Two Comparators with Programmable Interrupt Priority
- Six Vectored, Z8 Priority Interrupts
- RAM and ROM Protect
- Clock Speed of 20.48 MHz
- Z89C00 Core 16-Bit Digital Signal Processor (DSP)
- 4K Words DSP Program ROM
- 512 Words On-Chip DSP RAM
- 8-Bit A/D Converter with up to 128 kHz Sample Rate
- 10-Bit PWM D/A Converter (4 kHz to 64 kHz)
- Two DSP Timers to Support Different A/D and D/A Sampling Rates (Automatic Triggering—A/D + D/A may be Asynchronous Triggered)
- Zero Overhead Hardware Looping
- 16-Bit Single Cycle DSP Instructions
- 16-Bit Single Cycle DSP Multiply/Accumulate
- Six-Level DSP Stack
- 24-Bit DSP ALU, Accumulator and Shifter
- Three Vectored DSP Interrupts
- Independent Z8 and DSP Operations
- IBM® PC-Based Development Tools

GENERAL DESCRIPTION

The Z89120/920 is a fully integrated, mixed signal, dual processor chip system designed for lower sample rates such as audio, telephone, security systems, modem, faxes, instrumentation, noise cancellation, and sonar. The I/O control processor is a Z8® with 24 Kbytes of program memory, two 8-bit counter timers, and up to 47 I/O pins. The DSP is a 16-bit processor with a 24-bit ALU and Accumulator, 512 x 16 bits of RAM, single cycle instructions, and 6K word program ROM memory. The chip also contains an 8-bit A/D converter with up to 128 kHz sample rate and 10-bit PWM D/A converter. The sampling rates for the converters are independently programmable. The precision of the 8-bit A/D may be extended by resampling the data at a lower rate in software. Separate power supply pins are provided to increase noise immunity. The A/D has external reference inputs wich may be connected to the Z8 ports to affect two-three extra bits of automatic gain control/dynamic range.

The Z8 and DSP processors are loosely coupled by mailbox registers and an interrupt system. DSP or Z8 programs may be directed by events in each other's domain.

The Z89920 is the ROMless version of the Z89120. In the Z89920, only the Z8 is ROMless, the DSP is not ROMless. The DSP's program memory is always the internal ROM.

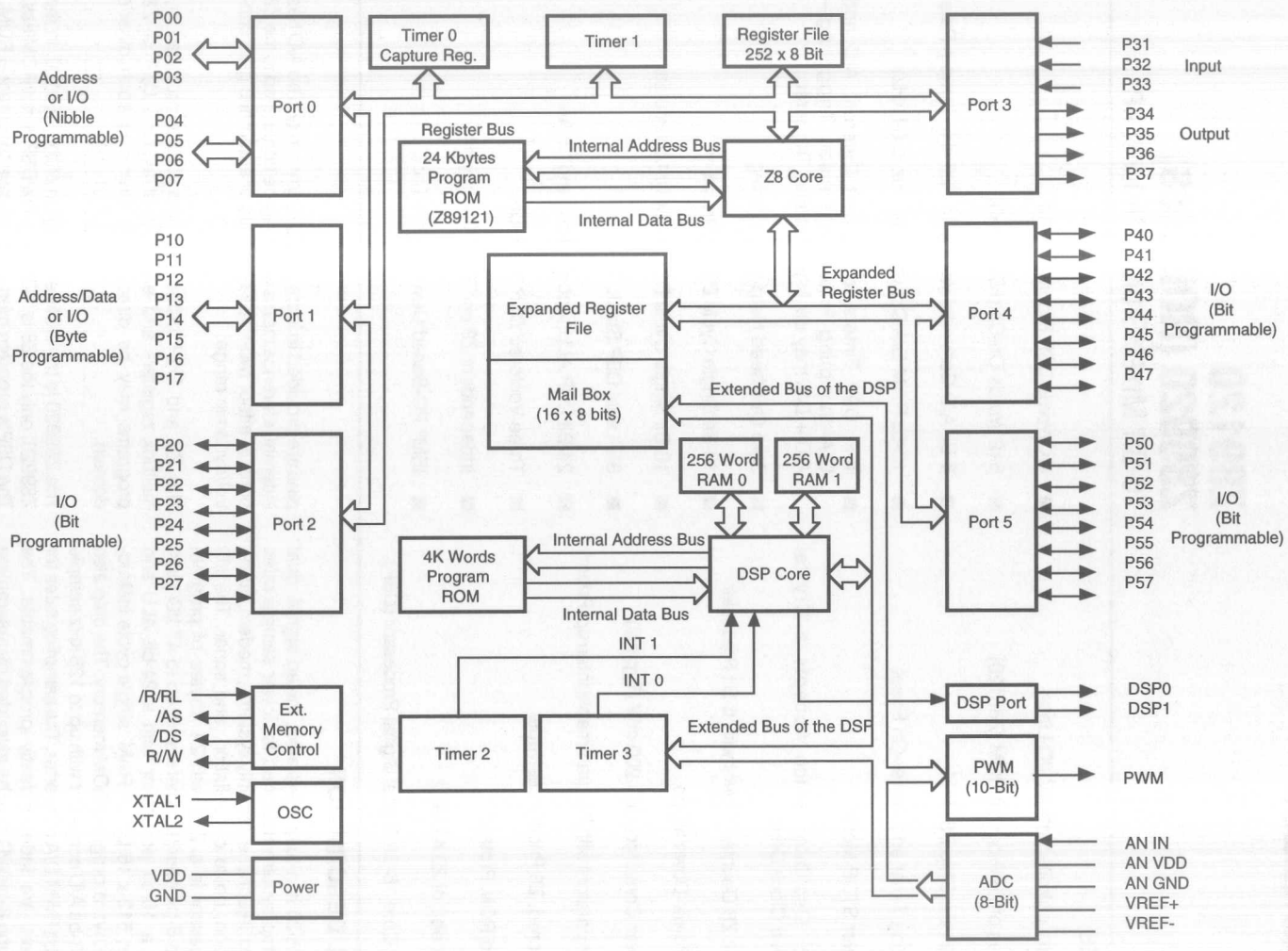


Figure 1. Functional Block Diagram

Z8® Core Processor

The Z8 is Zilog's 8-bit microcontroller core with an Expanded Register File to allow access to register-mapped peripheral and I/O circuits and the DSP. The Z8 offers a flexible I/O scheme, an efficient register and address space structure, and a number of ancillary features. The Z8 pipelined instructions and register file memory provide improved programming efficiency.

For applications demanding powerful I/O capabilities, the Z89120/920 fulfills this with 47 pins dedicated to input and output. These lines are grouped into five ports. Each port is configurable under software control to provide timing, status signals and parallel I/O with or without handshake.

There are four basic memory resources for the Z8 that are available to support a wide range of configurations: Program Memory, Register File, Data Memory, and Expanded Register File. The Z8 core processor is characterized by an efficient register file that allows any of 256 on-board data and control registers to be the source and/or the destination of almost any instruction. Traditional microprocessor Accumulator bottlenecks are eliminated.

The Register File is composed of 236 bytes of general-purpose registers, four I/O port registers, and 15 control and status registers. The Expanded Register File consists of control registers, mailbox registers, two additional timing registers and data registers. Peripheral control registers and an additional Port 4 and Port 5 are mapped into the expanded-register file to further enhance the system performance and code efficiency.

To unburden the software from supporting the real-time problems, such as counting/timing and data communication, the Z8 offers two on-chip counter/timers with a large number of user selectable modes.

Watch-Dog Timer (WDT) and Stop-Mode Recovery (SMR) features are software driven by setting specific bits in control registers.

STOP and HALT instructions support reduced power operation. The low power STOP mode allows parameter information to be stored in the register file if power fails. An external capacitor or battery retains power to the device.

DSP Coprocessor

The DSP coprocessor is a second generation, 16-bit two's complement CMOS Digital Signal Processor (DSP). Most instructions, including multiply and accumulate, are accomplished in a single clock cycle. The processor contains two on-chip data RAM blocks of 256 words, a 4K word program ROM, 24-bit ALU, 16 x 16 multipiler, 24-bit Accumulator, shifter, six-level stack, three vectored interrupts and two inputs for conditional program jumps. Each RAM block contains a set of four pointers which may be incremented or decremented automatically to affect hardware looping without software overhead. The data RAMs can be simultaneously addressed and loaded to the multiplier for a true single cycle scalar multiply.

Four extended DSP registers are mapped into the expanded register file of the Z8. Communication between the Z8 and the DSP occurs through those common registers which form the mailbox registers.

Analog Interface

The analog signal is generated by a 10-bit resolution Pulse Width Modulator D/A converter. The PWM output is a digital signal with CMOS output levels. The output signal has a resolution of 1 in 1024 with a sampling rate of 16 kHz (XTAL = 20.48 MHz). The sampling rate can be changed under software control and can be set at 4, 10, 16, and 64 kHz. The dynamic range of the PWM is from 0 to 4V.

An 8-bit resolution half flash A/D converter is provided. The conversion is conducted with a sampling frequency of 8, 16, 32, 64, or 128 kHz. (XTAL = 20.48 MHz) in order to provide oversampling. The input signal maybe up to 5V peak to peak. Normally input signals are level shifted to 2.5V with a ±2.5V deviation.

The reference voltages for the A/D converter can be adjusted by the Z8 by external connection to provide automatic gain control.

Two additional timers (Timer2 and Timer3) have been added to support different sampling rates for the A/D and D/A converters. These timers are free running counters that divide the crystal frequency.

Notes:
All signals with a preceding front slash, "/", are active Low, e.g., B/W (WORD is active Low); /B/W (BYTE is active Low, only).

Power connections follow conventional descriptions below:

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

PIN DESCRIPTION

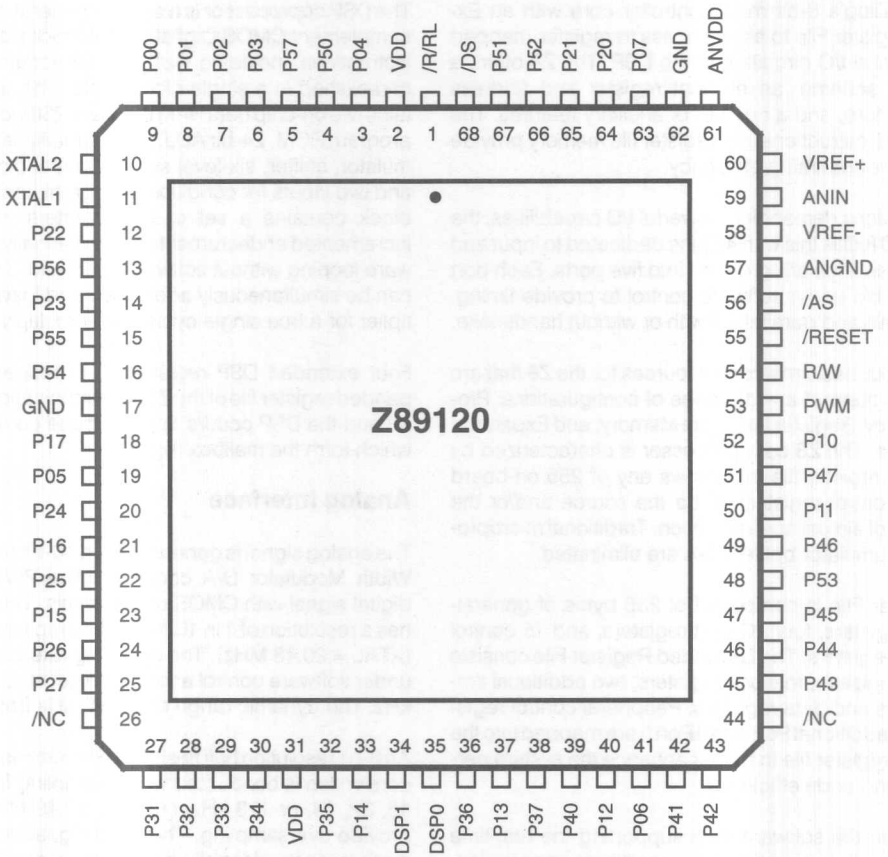


Figure 2. Z89120 68-Pin Plastic Leaded Chip Carrier, Pin Assignments

Table 1. Z89120 68-Pin Plastic Leaded Chip Carrier, Pin Identification

Pin #	Symbol	Function	Direction
1	/R/RL	ROM/ROMless	Control Input
2	V_{DD}	Power Supply	
3	P04	Port 0, Bit 4	Input/Output
4	P50	Port 5, Bit 0	Input/Output
5	P57	Port 5, Bit 7	Input/Output
6	P03	Port 0, Bit 3	Input/Output
7	P02	Port 0, Bit 2	Input/Output
8	P01	Port 0, Bit 1	Input/Output
9	P00	Port 0, Bit 0	Input/Output
10	XTAL2	Crystal Oscillator Clock	Output
11	XTAL1	Crystal Oscillator Clock	Input
12	P22	Port 2, Bit 2	Input/Output
13	P56	Port 5, Bit 6	Input/Output
14	P23	Port 2, Bit 3	Input/Output
15	P55	Port 5, Bit 5	Input/Output
16	P54	Port 5, Bit 4	Input/Output
17	GND	Ground	
18	P17	Port 1, Bit 7	Input/Output
19	P05	Port 0, Bit 5	Input/Output
20	P24	Port 2, Bit 4	Input/Output
21	P16	Port 1, Bit 6	Input/Output
22	P25	Port 2, Bit 5	Input/Output
23	P15	Port 1, Bit 5	Input/Output
24	P26	Port 2, Bit 6	Input/Output
25	P27	Port 2, Bit 7	Input/Output
26	NC	Not Connected	
27	P31	Port 3, Bit 1	Input
28	P32	Port 3, Bit 2	Input
29	P33	Port 3, Bit 3	Input
30	P34	Port 3, Bit 4	Output
31	V_{DD}	Power Supply	
32	P35	Port 3, Bit 5	Output
33	P14	Port 1, Bit 4	Input/Output
34	DSP1	DSP User Output 1	Output

Pin #	Symbol	Function	Direction
35	DSP0	DSP User Output 0	Output
36	P36	Port 3, Bit 7	Output
37	P13	Port 1, Bit 3	Input/Output
38	P37	Port 3, Bit 7	Output
39	P40	Port 4, Bit 0	Input/Output
40	P12	Port 1, Bit 2	Input/Output
41	P06	Port 0, Bit 6	Input/Output
42	P41	Port 4, Bit 1	Input/Output
43	P42	Port 4, Bit 2	Input/Output
44	NC	Not Connected	
45	P43	Port 4, Bit 3	Input/Output
46	P44	Port 4, Bit 4	Input/Output
47	P45	Port 4, Bit 5	Input/Output
48	P53	Port 5, Bit 3	Input/Output
49	P46	Port 4, Bit 6	Input/Output
50	P11	Port 1, Bit 1	Input/Output
51	P47	Port 4, Bit 7	Input/Output
52	P10	Port 1, Bit 0	Input/Output
53	PWM	Pulse Width Modulator	Output
54	R/W	Read/Write	Output
55	/RESET	Reset	Input
56	/AS	Address Strobe	Output
57	AN_{GND}	Analog Ground	
58	V_{REF-}	Analog Voltage Ref.	Input
59	AN_{IN}	Analog Input	Input
60	V_{REF+}	Analog Voltage Ref.	Input
61	ANV_{DD}	Analog Power Supply	
62	GND	Ground	
63	P07	Port 0, Bit 7	Input/Output
64	P20	Port 2, Bit 0	Input/Output
65	P21	Port 2, Bit 1	Input/Output
66	P52	Port 5, Bit 2	Input/Output
67	P51	Port 5, Bit 1	Input/Output
68	/DS	Data Strobe	Output

PIN DESCRIPTION (Continued)

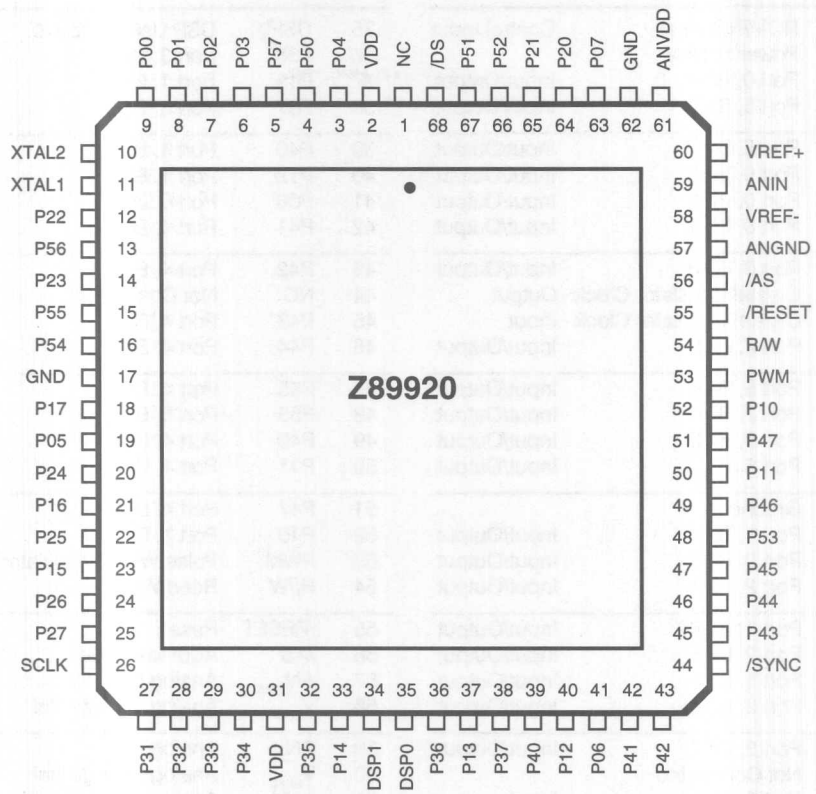


Figure 3. Z89920 68-Pin Plastic Leaded Chip Carrier, Pin Assignments

Table 2. Z89920 68-Pin Plastic Leaded Chip Carrier, Pin Identification

Pin #	Symbol	Function	Direction
1	NC	Not Connected	
2	V_{DD}	Power Supply	
3	P04	Port 0, Bit 4	Input/Output
4	P50	Port 5, Bit 0	Input/Output
5	P57	Port 5, Bit 7	Input/Output
6	P03	Port 0, Bit 3	Input/Output
7	P02	Port 0, Bit 2	Input/Output
8	P01	Port 0, Bit 1	Input/Output
9	P00	Port 0, Bit 0	Input/Output
10	XTAL2	Crystal Oscillator Clock	Output
11	XTAL1	Crystal Oscillator Clock	Input
12	P22	Port 2, Bit 2	Input/Output
13	P56	Port 5, Bit 6	Input/Output
14	P23	Port 2, Bit 3	Input/Output
15	P55	Port 5, Bit 5	Input/Output
16	P54	Port 5, Bit 4	Input/Output
17	GND	Ground	
18	P17	Port 1, Bit 7	Input/Output
19	P05	Port 0, Bit 5	Input/Output
20	P24	Port 2, Bit 4	Input/Output
21	P16	Port 1, Bit 6	Input/Output
22	P25	Port 2, Bit 5	Input/Output
23	P15	Port 1, Bit 5	Input/Output
24	P26	Port 2, Bit 6	Input/Output
25	P27	Port 2, Bit 7	Input/Output
26	SCLK	System Clock	Output
27	P31	Port 3, Bit 1	Input
28	P32	Port 3, Bit 2	Input
29	P33	Port 3, Bit 3	Input
30	P34	Port 3, Bit 4	Output
31	V_{DD}	Power Supply	
32	P35	Port 3, Bit 5	Output
33	P14	Port 1, Bit 4	Input/Output
34	DSP1	DSP User Output 1	Output
35	DSP0	DSP User Output 0	Output
36	P36	Port 3, Bit 7	Output
37	P13	Port 1, Bit 3	Input/Output
38	P37	Port 3, Bit 7	Output
39	P40	Port 4, Bit 0	Input/Output
40	P12	Port 1, Bit 2	Input/Output
41	P06	Port 0, Bit 6	Input/Output
42	P41	Port 4, Bit 1	Input/Output
43	P42	Port 4, Bit 2	Input/Output
44	/SYNC	Synchronize Pin	Output
45	P43	Port 4, Bit 3	Input/Output
46	P44	Port 4, Bit 4	Input/Output
47	P45	Port 4, Bit 5	Input/Output
48	P53	Port 5, Bit 3	Input/Output
49	P46	Port 4, Bit 6	Input/Output
50	P11	Port 1, Bit 1	Input/Output
51	P47	Port 4, Bit 7	Input/Output
52	P10	Port 1, Bit 0	Input/Output
53	PWM	Pulse Width Modulator	Output
54	R/W	Read/Write	Output
55	/RESET	Reset	Input
56	/AS	Address Strobe	Output
57	AN_{GND}	Analog Ground	
58	V_{REF-}	Analog Voltage Ref.	Input
59	AN_{IN}	Analog Input	Input
60	V_{REF+}	Analog Voltage Ref.	Input
61	ANV_{DD}	Analog Power Supply	
62	GND	Ground	
63	P07	Port 0, Bit 7	Input/Output
64	P20	Port 2, Bit 0	Input/Output
65	P21	Port 2, Bit 1	Input/Output
66	P52	Port 5, Bit 2	Input/Output
67	P51	Port 5, Bit 1	Input/Output
68	/DS	Data Strobe	Output

PIN FUNCTIONS

/RESET (input, active Low). Initializes the MCU. Reset is accomplished either through Power-On Reset (POR), WDT reset, SMR or external reset. During POR and WDT reset, the internally generated reset is driving the reset pin Low for the POR time. Any devices driving the reset line must be open drain to avoid damage from a possible conflict during reset conditions. Pull-up is provided internally. A /RESET signal resets both the Z8 and the DSP.

For the Z8:
After the POR time, /RESET is a Schmitt-triggered input. To avoid asynchronous and noisy reset problems, the Z8 is equipped with a reset filter of four external clocks (4TpC). If the external reset signal is less than 4TpC in duration, no reset occurs. On the fifth clock after the reset is detected, an internal RST signal is latched and held for an internal register count of 18 external clocks, or for the duration of the external reset, whichever is longer. Program execution begins at location 000CH (Hexadecimal), 5-10 TpC cycles after the /RESET is released. For Power-On Reset, the typical reset time is 5 ms. The Z8 does not reset WDT, SMR, P2M, and P3M registers on a Stop-Mode Recovery operation.

For the DSP:
A low level on the /RESET pin generates an internal reset signal. The /RESET signal must be kept low for at least one clock cycle. The CPU will push the contents of the PC onto the stack and then fetch a new Program Counter (PC) value from program memory address 0FFCH after the reset signal is released.

/R/RL *ROM/ROMless* (input, active Low). This pin, when connected to V_{CC}, disables the internal Z8 ROM only and forces the device to function as a Z89920 ROMless. (Note that when pulled Low to GND, the Z89120 functions normally as the ROM version). The DSP can not be configured as ROMless. This is available only on the Z89120.

R/W *Read/Write* (output, write Low). The R/W signal defines the signal flow when the Z8 is reading or writing to external program or data memory. The Z8 is reading when this pin is High and writing when this pin is Low.

/AS *Address Strobe* (output, active Low). Address Strobe is pulsed once at the beginning of each machine cycle. Address output is through Port 0/Port 1 for all external programs. Memory address transfers are valid at the trailing edge of /AS. Under program control, /AS is placed in the high-impedance state along with Ports 0 and 1, Data Strobe, and Read/Write.

/DS *Data Strobe* (output, active Low). Data Strobe is activated once for each external memory transfer. For read operations, data must be available prior to the trailing edge of /DS. For write operations, the falling edge of /DS indicates that output data is valid.

XTAL1 *Crystal 1* (time-based input). This pin connects a parallel-resonant crystal, ceramic resonator, LC, RC network or an external single-phase clock to the on-chip oscillator input.

XTAL2 *Crystal 2* (time-based output). This pin connects a parallel-resonant, crystal, ceramic resonant, or LC network to the on-chip oscillator output.

DSP0 (output). DSP0 is a general purpose output pin connected to bit 6 of the Analog Control Register (DSP EXT4). This bit has no special significance and may be used to output data by writing to bit 6 of the ACR.

DSP1 (output). DSP1 is a general purpose output pin connected to bit 7 of the Analog Control Register (DSP EXT4). This bit has no special significance and may be used to output data by writing to bit 7 of the ACR.

SCLK *System Clock* (output). SCLK outputs the internal system clock. This pin is only available on the Z89920.

/SYNC *Synchronize* (output). This signal indicates the last clock cycle of the currently executing Z8 instruction. This pin is only available on the Z89920.

PWM *Pulse Width Modulator* (output). The PWM is a 10-bit resolution D/A converter. This output is a digital signal with CMOS output levels.

AN_{IN} (input). Analog input for the A/D converter. Signal range is AN_{GND} to AN_{VDD}.

ANV_{CC}. Analog power supply for the A/D and D/A converters.

AN_{GND}. Analog ground for the A/D converter.

V_{REF+} (input). Reference voltage (High) for the A/D converter.

V_{REF-} (input). Reference voltage (Low) for the A/D converter.

V_{CC}. Digital power supply for the Z89120/920.

GND. Digital ground for the Z89120/920.

Port 0 (P07-P00). Port 0 is an 8-bit, bidirectional, CMOS compatible port. These eight I/O lines are configured under software control as a nibble I/O port, or as an address port for interfacing external memory. The input buffers are Schmitt-triggered and the output drivers are push-pull. Port 0 is placed under handshake control. In this configuration, Port 3, lines P32 and P35 are used as the handshake control /DAV0 and RDY0. Handshake signal direction is dictated by the I/O direction to Port 0 of the upper nibble P07-P04. The lower nibble must have the same direction as the upper nibble.

The Auto Latch on Port 0 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1, cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

For external memory references, Port 0 provides address bits A11-A8 (lower nibble) or A15-A8 (lower and upper nibble) depending on the required address space. If the address range requires 12 bits or less, the upper nibble of Port 0 can be programmed independently as I/O while the lower nibble is used for addressing. If one or both nibbles are needed for I/O operation, they are configured by writing to the Port 0 mode register.

In ROMless mode, after a hardware reset, Port 0 is configured as address lines A15-A8, and extended timing is set to accommodate slow memory access. The initialization routine can include reconfiguration to eliminate this extended timing mode. (In ROM mode, Port 0 is defined as input after reset.)

Port 0 is set in the high-impedance mode if selected as an address output state along with Port 1 and the control signals /AS, /DS and R/W (Figure 4).

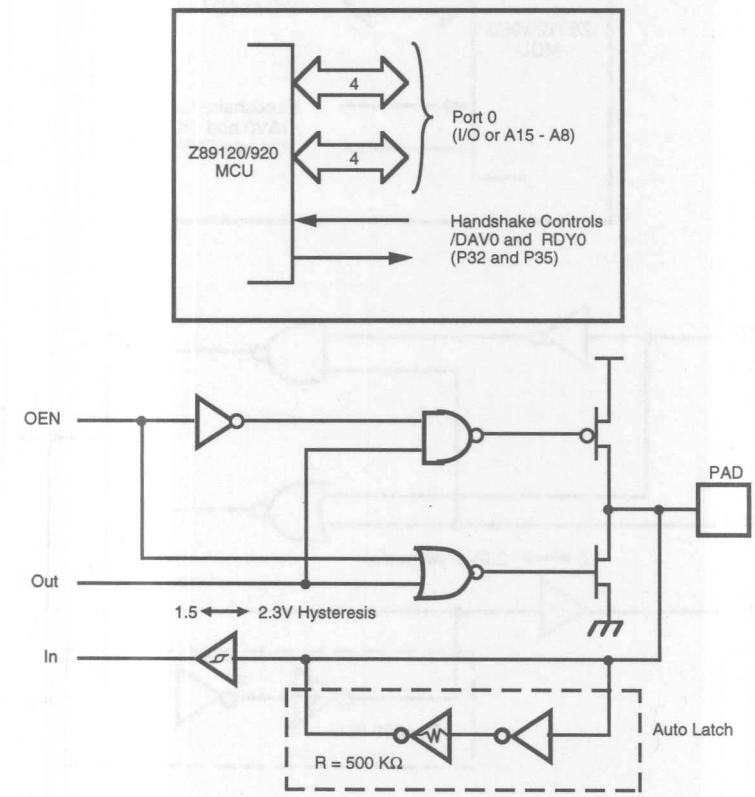


Figure 4. Port 0 Configuration

PIN FUNCTIONS (Continued)

Port 1 (P17-P10). Port 1 is an 8-bit, bidirectional, CMOS compatible port (Figure 5). It has multiplexed Address (A7-A0) and Data (D7-D0) ports. These eight I/O lines are programmed as inputs or outputs, or can be configured under software control as an Address/Data port for interfacing external memory. The input buffers are Schmitt-triggered and the output drivers are push-pull.

Port 1 may be placed under handshake control. In this configuration, Port 3, lines P33 and P34 are used as the handshake controls RDY1 and /DAV1 (Ready and Data Available). Memory locations greater than 24575 (in ROM mode) are referenced through Port 1. To interface external memory, Port 1 must be programmed for the multiplexed Address/Data mode. If more than 256 external locations are required, Port 0 outputs the additional lines.

Port 1 can be placed in the high-impedance state along with Port 0, /AS, /DS and R/W, allowing the Z89120/920 to share common resources in multiprocessor and DMA applications.

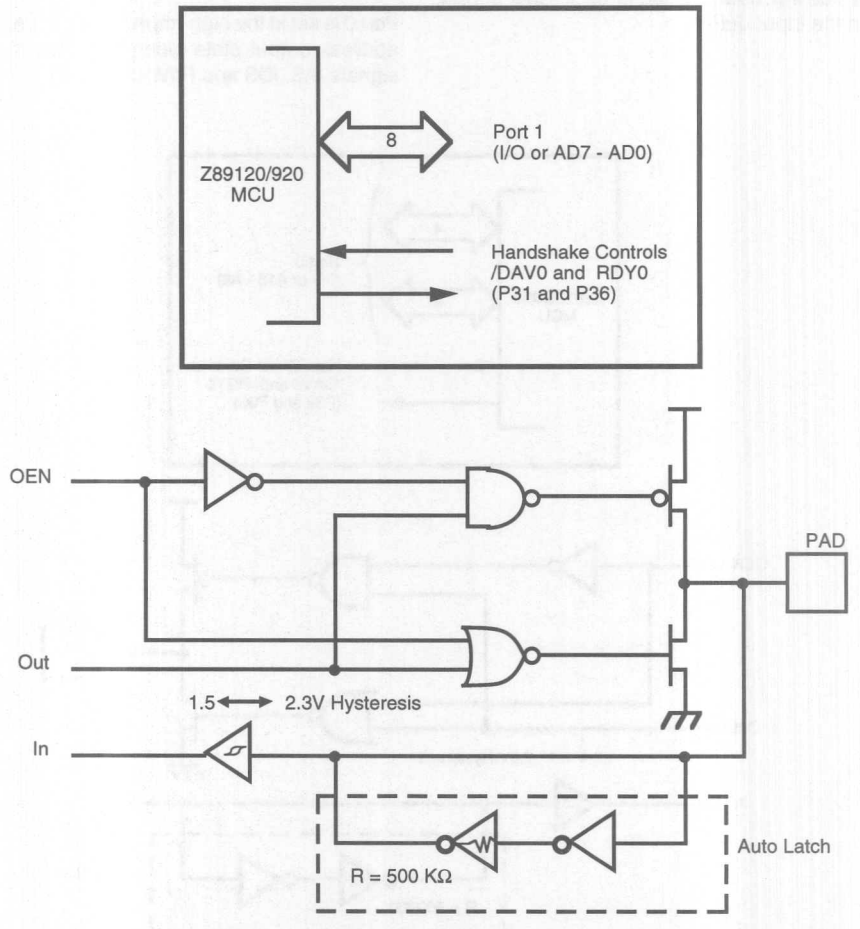


Figure 5. Port 1 Configuration

Port 2 (P27-P20). Port 2 is an 8-bit, bidirectional, CMOS compatible I/O port. These eight I/O lines are independently configured under software control as an input or output. Port 2 is always available for I/O operation. The input buffers are Schmitt-triggered. Bits programmed as outputs may be globally programmed as either push-pull or open-drain.

Port 2 may be placed under handshake control. In this configuration, Port 3 lines P31 and P36 are used as the handshake controls lines /DAV2 and RDY2. The handshake signal assignment for Port 3 lines P31 and P36 is dictated by the direction (input or output) assigned to bit 7, Port 2 (Figure 6).

Port 26 can be configured in DSP software to activate DSP INT0.

The Auto Latch on Port 2 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1, cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

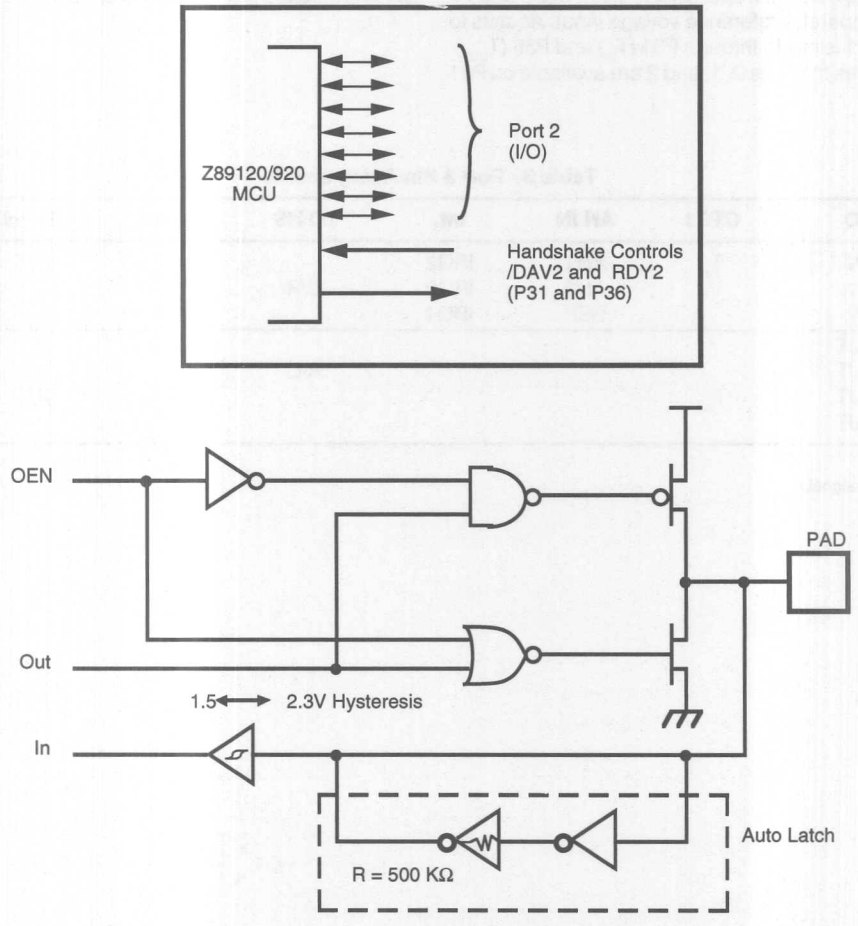


Figure 6. Port 2 Configuration

PIN FUNCTIONS (Continued)

Port 3 (P37-P31). Port 3 is a 7-bit, CMOS compatible port with three fixed inputs (P33-P31) and four fixed outputs (P37-P34). It is configured under software control for input/output, counter/timers, interrupt, and port handshakes. Pins P31, P32, and P33 are standard CMOS inputs; outputs are push-pull.

Two on-board comparators can process analog signals on P31 and P32 with reference to the voltage on P33. The analog function is enabled by programming the Port 3 Mode Register (bit 1). Port 3, pin 3 is a falling edge interrupt input. P31 and P32 are programmable as rising, falling or both edge-triggered interrupts (IRQ register bits 6 and 7). P33 is the comparator reference voltage input. Access to Counter/Timer1 is made through P31 (T_{IN}) and P36 (T_{OUT}). Handshake lines for Ports 0, 1, and 2 are available on P31 through P36.

Port 3 also provides the following control functions: handshake for Ports 0, 1, and 2 (/DAV and RDY); three external interrupt request signals (IRQ3-IRQ1); timer input and output signals (T_{IN} and T_{OUT}); Data Memory Select (/DM); (Figure 7).

Comparator Inputs. Port 3, Pins P31 and P32 each have a comparator front end. The comparator reference voltage, pin P33, is common to both comparators. In analog mode, the P31 and P32 are the positive inputs to the comparators and P33 is the reference voltage supplied to both comparators. In digital mode, pin P33 can be used as a P33 register input or IRQ1 source.

Table 3. Port 3 Pin Assignments

Pin	I/O	CTC1	AN IN	Int.	P0 HS	P1HS	P2 HS	EXT
P31	IN	T_{in}	AN1	IRQ2			D/R	
P32	IN		AN2	IRQ0	D/R			
P33	IN		REF	IRQ1		D/R		
P34	OUT					R/D		/DM
P35	OUT				R/D			
P36	OUT	T_{out}					R/D	
P37	OUT							

Notes:
HS = Handshake Signals
D = DAV
R = RDY

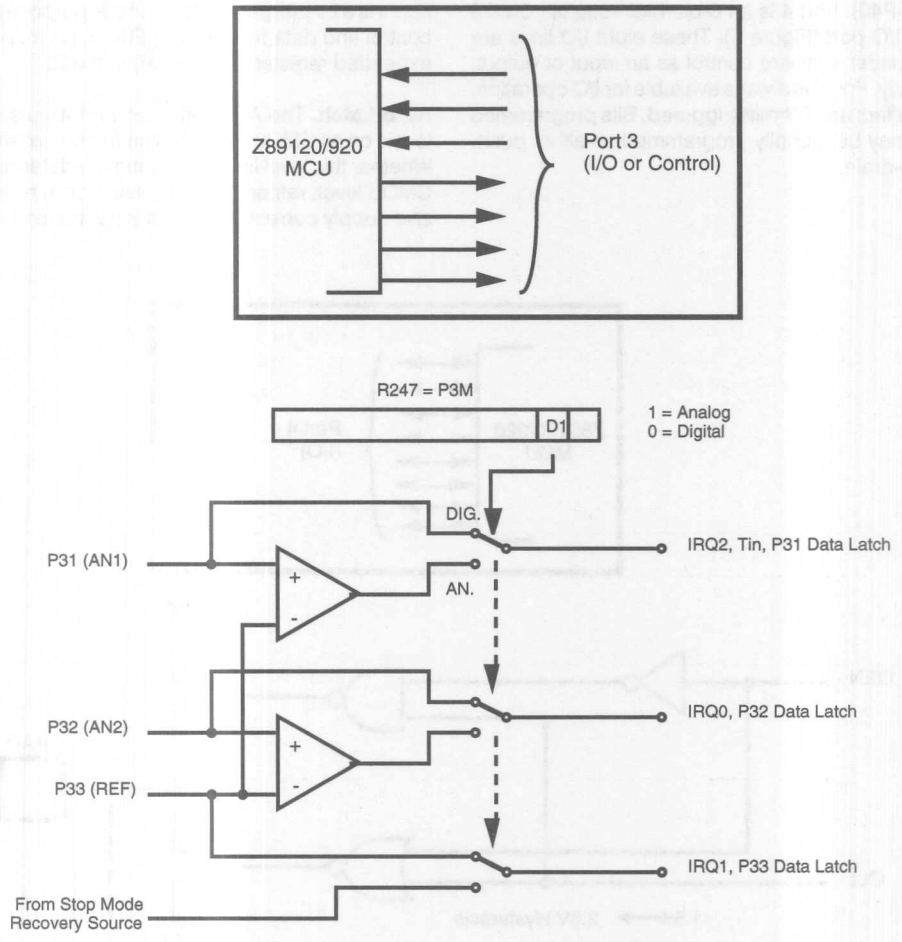


Figure 7. Port 3 Configuration

PIN FUNCTIONS (Continued)

Port 4 (P47-P40). Port 4 is an 8-bit, bidirectional, CMOS compatible I/O port (Figure 8). These eight I/O lines are configured under software control as an input or output, independently. Port 4 is always available for I/O operation. The input buffers are Schmitt-triggered. Bits programmed as outputs may be globally programmed as either push-pull or open-drain.

Port 4 is a bit programmable general purpose I/O port. The control and data registers for Port 4 are mapped into the expanded register file (Bank F) of the Z8.

Auto Latch. The Auto Latch on Port 4 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1, cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

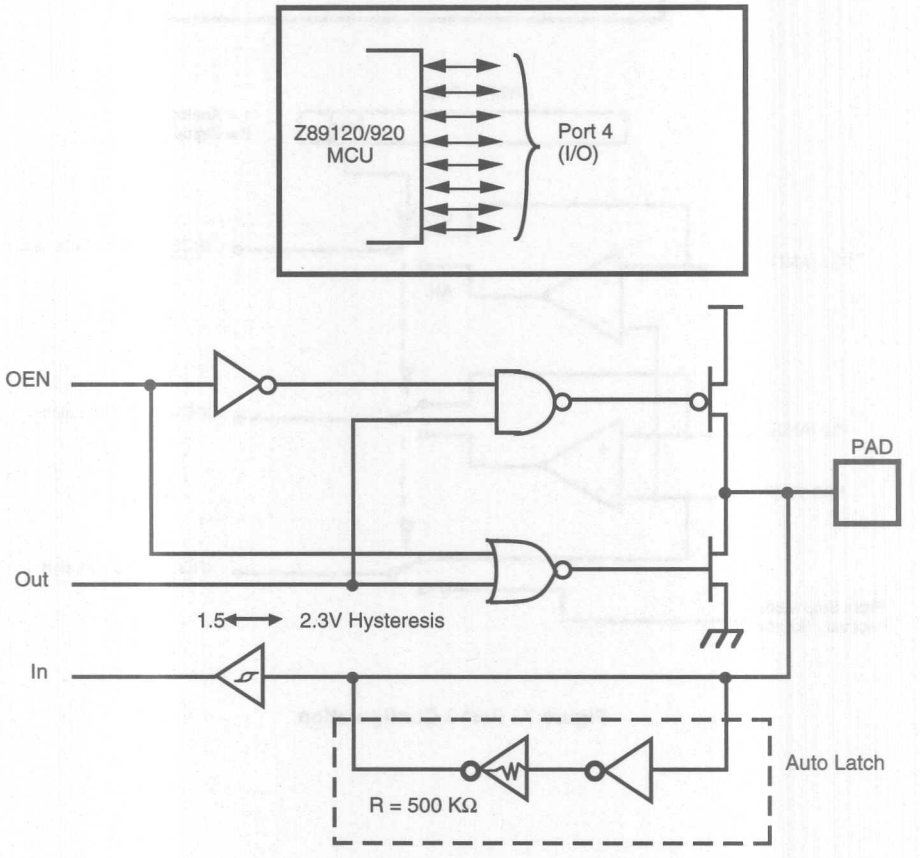


Figure 8. Port 4 Configuration

Port 5 (P57-P50). Port 5 is an 8-bit, bidirectional, CMOS compatible I/O port (Figure 9). These eight I/O lines are configured under software control as an input or output, independently. Port 5 is always available for I/O operation. The input buffers are Schmitt-triggered. Bits programmed as outputs may be globally programmed as either push-pull or open-drain.

Port 5 is a bit programmable general purpose I/O port. The control and data registers for Port 5 are mapped into the expanded register file (Bank F) of the Z8.

Auto Latch. The Auto Latch on Port 5 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1, cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

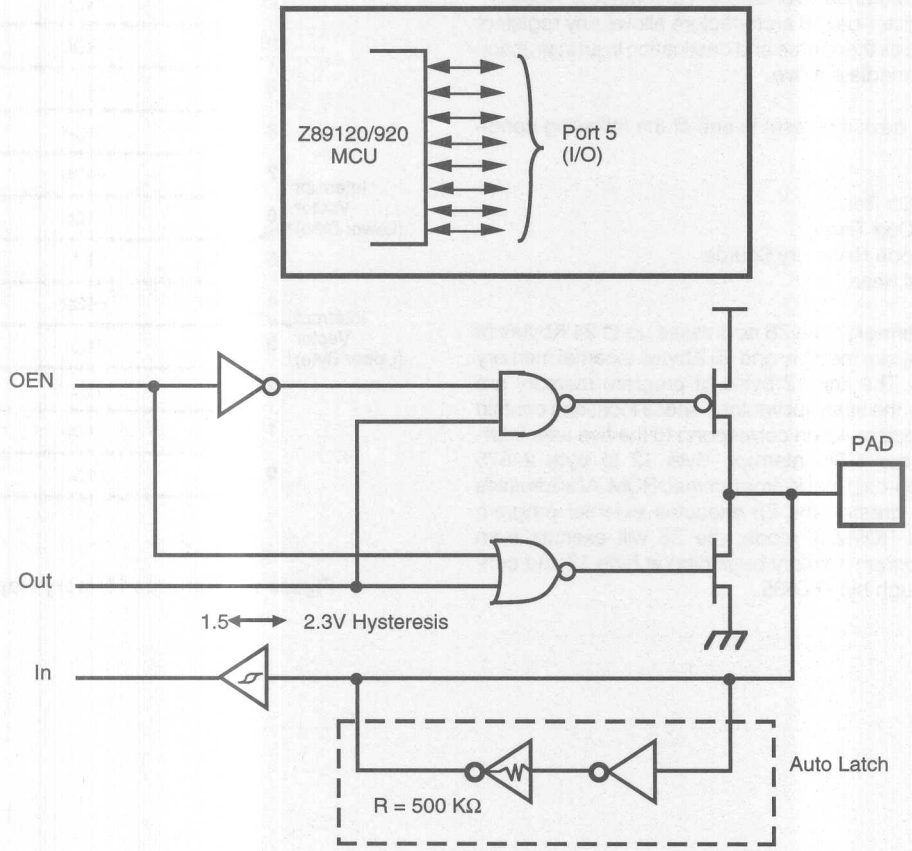


Figure 9. Port 5 Configuration

Z8® FUNCTIONAL DESCRIPTION

The Z8 CCP core incorporates special functions to enhance the Z8's performance in control applications.

Pipelined Instructions. The Z8 instructions (see page 73) are comprised of two parts, an instruction fetch and execute part. The instructions typically take between six and ten cycles to fetch and five cycles to execute. Five cycles of the next instruction fetch may be overlapped with five cycles of the current instruction execution. This improves performance over sequential methods. Additionally, the register-based archetecture allows any registers to be picked as the source and destination in an instruction saving intermediate move.

Reset. The device is reset in one of the following conditions:

- Power-On Reset
- Watch-Dog Timer
- Stop-Mode Recovery Source
- External Reset

Program Memory. The Z8 addresses up to 24 Kbytes of internal program memory and 40 Kbytes external memory (Figure 10). The first 12 bytes of program memory are reserved for the interrupt vectors. These locations contain six 16-bit vectors which correspond to the five user interrupts and one DSP interrupt. Byte 12 to byte 24575 consists of on-chip mask-programmed ROM. At addresses 24576 and greater, the Z8 executes external program memory. In ROMless mode, the Z8 will execute from external program memory beginning at byte 12 and continuing through byte 65535.

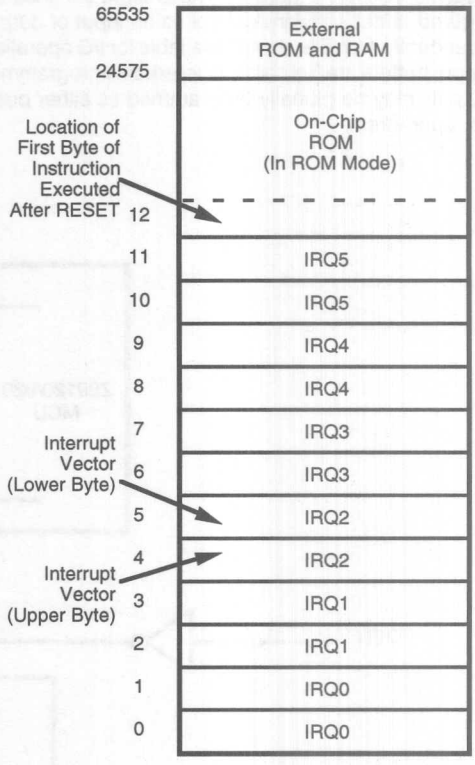


Figure 10. Program Memory Map

ROM Protect. The 24 Kbytes of internal program memory for the Z8 is mask programmable. A ROM Protect feature prevents "dumping" of the ROM contents of Program Memory by inhibiting execution of LDC, LDCI, LDE, and LDEI instructions. The ROM Protect option is mask-programmable, to be selected by the customer at the time when the ROM code is submitted.

Data Memory (/DM). In ROM mode, the Z8 can address up to 40 Kbytes of external data memory beginning at location 24576 (Figure 11). In ROMless mode, the Z8 can address the full 64 Kbytes of external data memory beginning at location 12. External data memory may be included with, or separated from, the external program memory space. /DM, an optional I/O function that can be programmed to appear on Port 34, is used to distinguish between data and program memory space (Table 3). The state of the /DM signal is controlled by the type of instruction being executed. An LDC opcode references PROGRAM (/DM inactive) memory, and an LDE instruction references data (/DM active Low) memory.

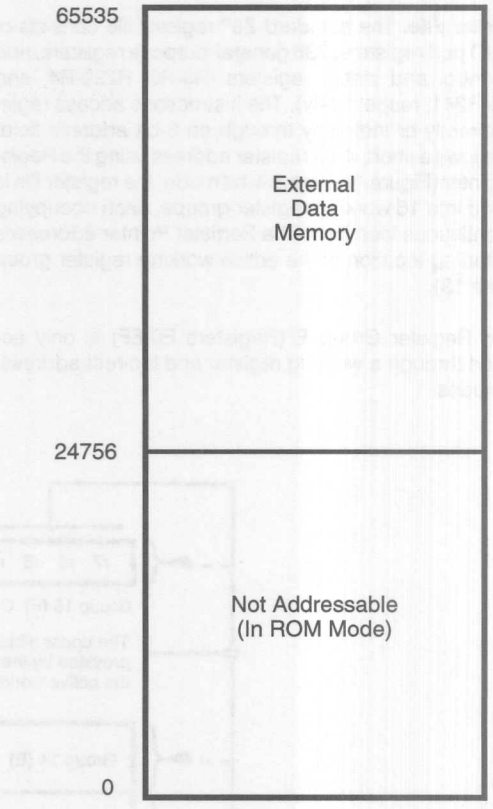


Figure 11. Data Memory Map

Z8 FUNCTIONAL DESCRIPTION (Continued)

Register File. The standard Z8® register file consists of four I/O port registers, 236 general-purpose registers, and 15 control and status registers (R3-R0, R239-R4, and R255-R241, respectively). The instructions access registers directly or indirectly through an 8-bit address field. This allows a short, 4-bit register address using the Register Pointer (Figure 12). In the 4-bit mode, the register file is divided into 16 working register groups, each occupying 16 continuous locations. The Register Pointer addresses the starting location of the active working register group (Figure 13).

Note: Register Group E (Registers E0-EF) is only accessed through a working register and indirect addressing modes.

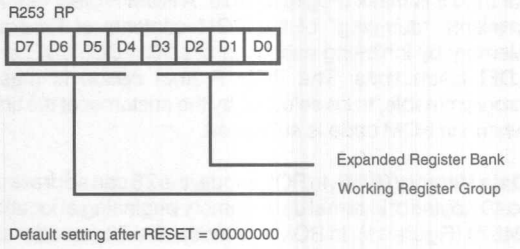


Figure 12. Register Pointer Register

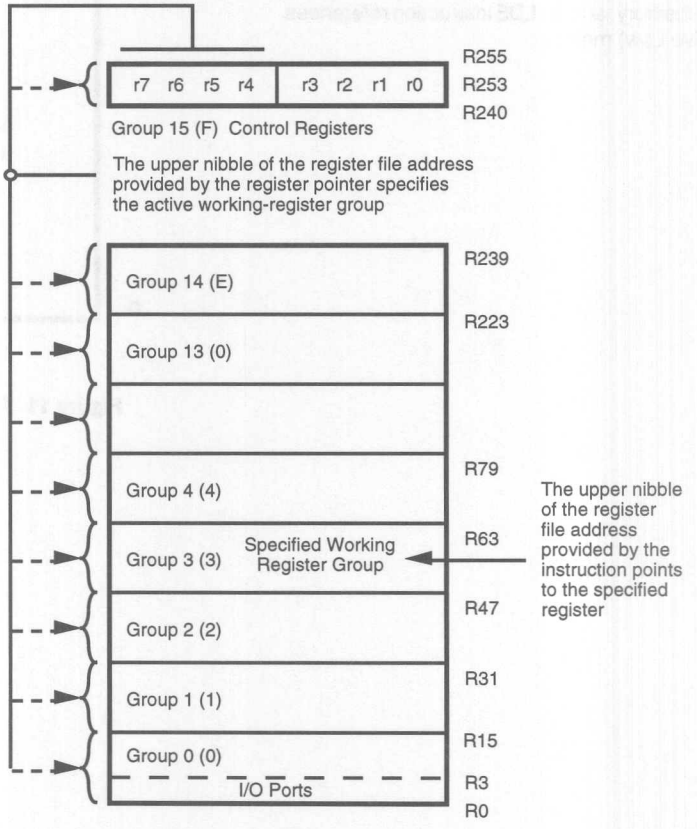


Figure 13. Register Pointer

RAM Protect. The upper portion of the Z8's RAM address spaces 80FH to EFH (excluding the control registers) are protected from reading and writing. The RAM Protect bit option is mask-programmable and is selected by the customer when the ROM code is submitted. After the mask option is selected, the user activates from the internal ROM code to turn off/on the RAM Protect by loading a bit D6 in the IMR register to either a 0 or a 1, respectively. A 1 in D6 indicates RAM Protect enabled.

Stack. The Z8's external data memory or the internal register file is used for the stack. The 16-bit Stack Pointer (R255-R254) is used for the external stack which can reside only from 24576 to 65535 in ROM mode or 0 to 65535 in ROMless mode. An 8-bit Stack Pointer (R255) is used for the internal stack that resides within the 236 general-purpose registers (R239-R4). SPH can be used as a general-purpose register when using internal stack only.

Expanded Register File. The register file on the Z8 has been expanded to allow for additional system control registers, and for mapping of additional peripheral devices along with I/O ports into the register address area. The Z8 register address space has now been implemented as 16 banks of 16 registers groups per bank (Figure 14). These register banks are known as the ERF (Expanded Register File). Bits 7-4 of register RP (Register Pointer) select the working register group. Bits 3-0 of register RP select the Expanded Register bank (Figure 14).

System configuration registers, Ports 4 and 5 mode registers, data registers and a DSP control register reside in Bank F of the Expanded Register File. Bank B of the Expanded Register File consists of the Mailbox Interface in which the Z8 and the DSP communicate. The rest of the Expanded Register is not physically implemented and is open for future expansion.

Z8 FUNCTIONAL DESCRIPTION (Continued)

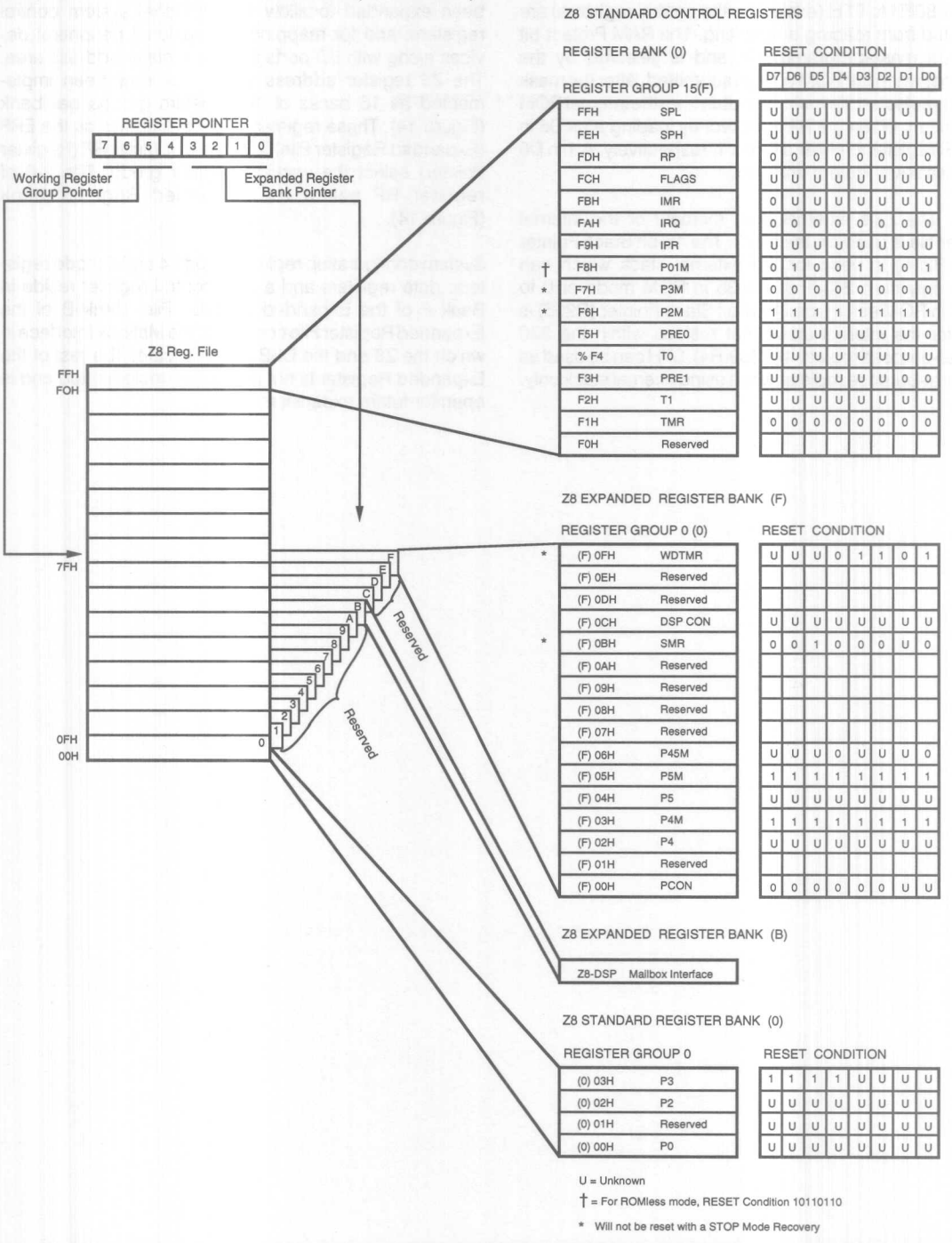


Figure 14. Expanded Register File Architecture

Interrupts. The Z8 has six different interrupts from six different sources. The interrupts are maskable and prioritized (Figure 15). The six sources are divided as follows; three sources are claimed by Port 3 lines P33-P31, two in counter/timers, and one by the DSP (Table 4). The Interrupt Mask Register globally or individually enables or disables the six interrupt requests.

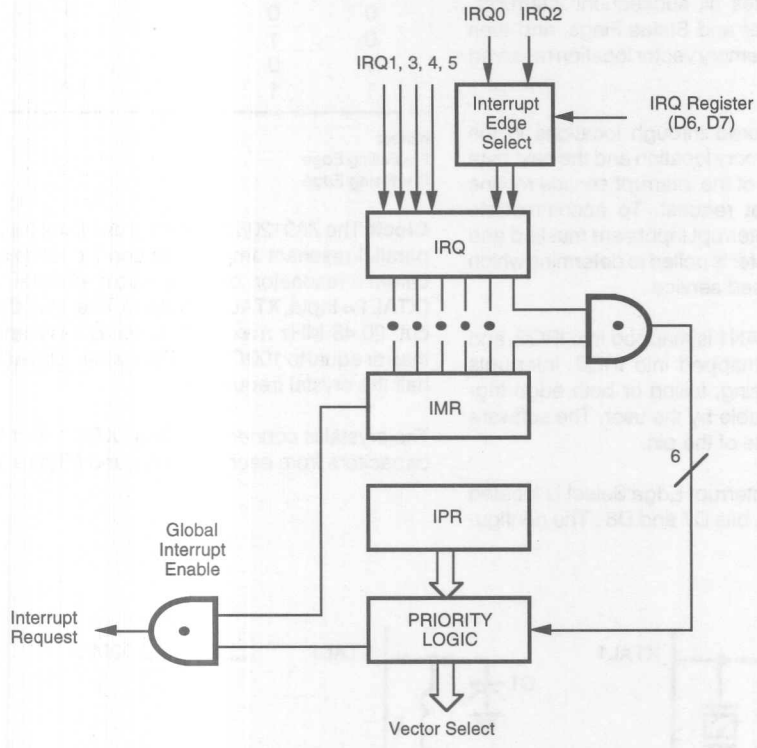


Figure 15. Interrupt Block Diagram

Table 4. Interrupt Types, Sources, and Vectors

Name	Source	Vector Location	Comments
IRQ0	/DAV0, P32, AN2	0, 1	External (P32), Programmable Rise or Fall Edge Triggered
IRQ1	/DAV1, P33	2, 3	External (P33), Fall Edge Triggered
IRQ2	/DAV2, P31, T_{IN}, AN2	4, 5	External (P31), Programmable Rise or Fall Edge Triggered
IRQ3	DSP	6, 7	Internal (DSP activated)
IRQ4	T0	8, 9	Internal
IRQ5	TI	10, 11	Internal

Z8 FUNCTIONAL DESCRIPTION (Continued)

When more than one interrupt is pending, priorities are resolved by a programmable priority encoder that is controlled by the Interrupt Priority Register. An interrupt machine cycle is activated when an interrupt request is granted. Thus, this disables all subsequent interrupts, saves the Program Counter and Status Flags, and then branches to the program memory vector location reserved for that interrupt.

All Z8 interrupts are vectored through locations in the program memory. This memory location and the next byte contain the 16-bit address of the interrupt service routine for that particular interrupt request. To accommodate polled interrupt systems, interrupt inputs are masked and the Interrupt Request Register is polled to determine which of the interrupt requests need service.

An interrupt resulting from AN1 is mapped into IRQ2, and an interrupt from AN2 is mapped into IRQ0. Interrupts IRQ2 and IRQ0 may be rising, falling or both edge triggered, and are programmable by the user. The software may poll to identify the state of the pin.

Programming bits for the Interrupt Edge Select is located in the IRQ Register (R250), bits D7 and D6 . The configuration is shown in Table 5.

Table 5. IRQ Register

IRQ		Interrupt Edge	
D7	D6	P31	P32
0	0	F	F
0	1	F	R
1	0	R	F
1	1	R/F	R/F

Notes:
F = Falling Edge
R = Rising Edge

Clock. The Z89120/920 on-chip oscillator has a high-gain, parallel-resonant amplifier for connection to a crystal, LC, ceramic resonator, or any suitable external clock source (XTAL1 = Input, XTAL2 = Output). The crystal should be AT cut, 20.48 MHz max., with a series resistance (RS) less than or equal to 100 Ohms. The system clock (SCLK) is one half the crystal frequency.

The crystal is connected across XTAL1 and XTAL2 using capacitors from each pin to ground (Figure 16).

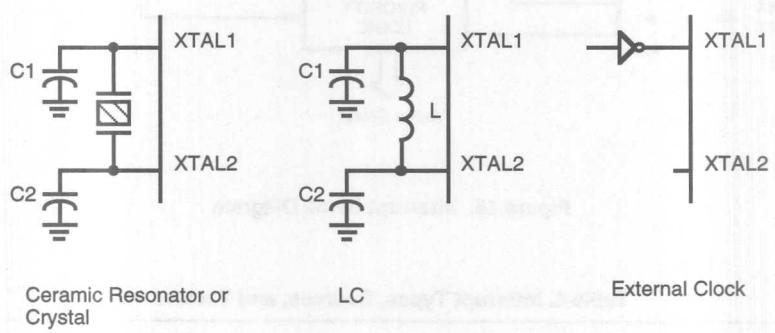


Figure 16. Oscillator Configuration

Counter/Timers. There are two 8-bit programmable counter/timers (T1-T0), each driven by its own 6-bit programmable prescaler. The T1 prescaler is driven by internal or external clock sources; however, the T0 prescaler is driven by the internal clock only (Figure 17).

The 6-bit prescalers can divide the input frequency of the clock source by any integer number from 1 to 64. Each prescaler drives its counter, which decrements the value (1 to 256) that has been loaded into the counter. When the counter reaches the end of the count, a timer interrupt request, IRQ4 (T0) or IRQ5 (T1), is generated.

The counters can be programmed to start, stop, restart to continue, or restart from the initial value. The counters can also be programmed to stop upon reaching zero (single pass mode) or to automatically reload the initial value and continue counting (modulo-n continuous mode).

The counters, but not the prescalers, are read at any time without disturbing their value or count mode. The clock source for T1 is user-definable and is either the internal microprocessor clock divided by four, or an external signal input through Port 31. The Timer Mode register configures the external timer input (P31) as an external clock, a trigger input that can be retriggerable or non-retriggerable, or as a gate input for the internal clock. The counter/timers can be cascaded by connecting the T0 output to the input of T1.

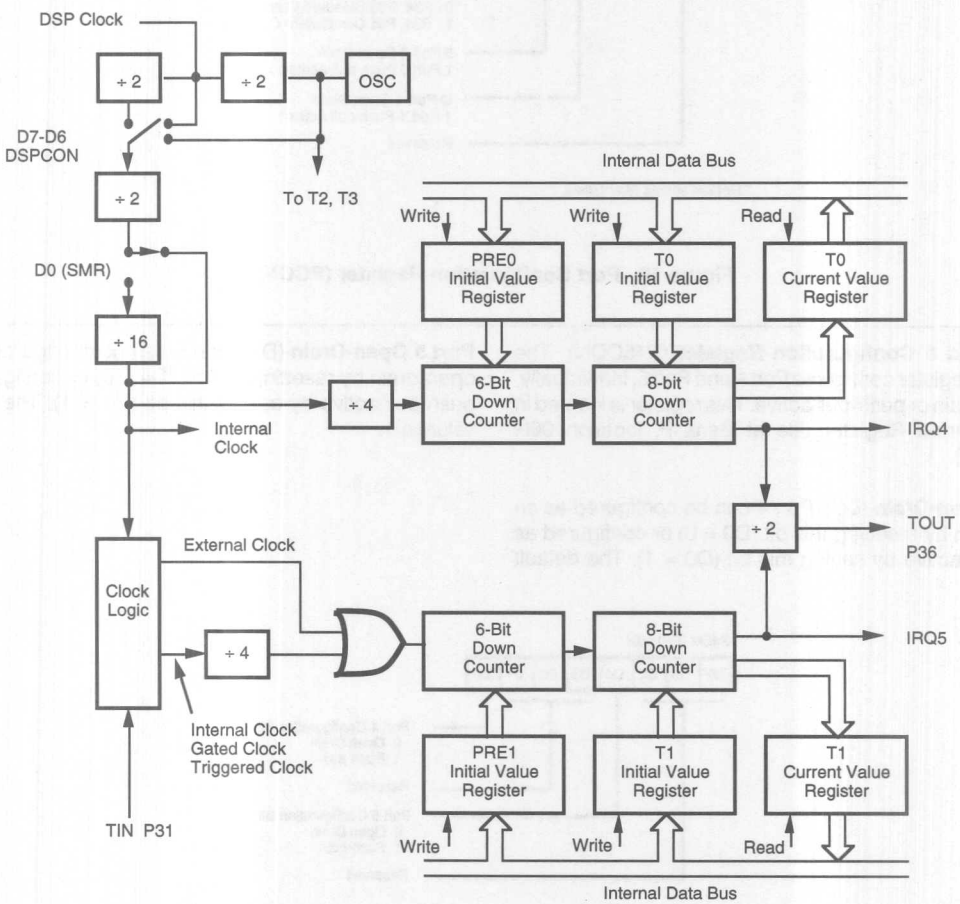


Figure 17. Counter/Timer Block Diagram

Z8 FUNCTIONAL DESCRIPTION (Continued)

Port Configuration Register (PCON). The PCON register configures the port individually; comparator output on Port 3, and open-drain on Port 0 and Port 1. The PCON register is located in the Expanded Register File at Bank F, location 00H (Figure 18).

Comparator Output Port 3 (D0). Bit 0 controls the comparator use in Port 3. A 1 in this location brings the comparator outputs to P34 and P35, and a 0 releases the port to its standard I/O configuration.

Port 0 Open-Drain (D1). Port 0 can be configured as an open-drain by resetting this bit (D1 = 0) or configured as push-pull active by setting this bit (D1 = 1). The default value is 1.

Port 1 Open-Drain (D2). Port 1 can be configured as an open-drain by resetting this bit (D2 = 0) or configured as push-pull active by setting this bit (D2 = 1). The default value is 1.

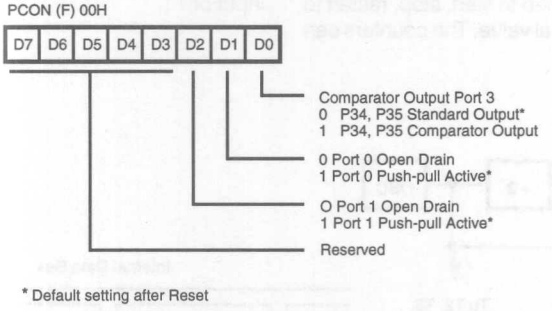


Figure 18. Port Configuration Register (PCON)

Port 4 and 5 Configuration Register (P45CON). The P45CON register configures Port 4 and Port 5, individually, to open-drain or push-pull active. This register is located in the Expanded Register File at Bank F, location 06H (Figure 18).

Port 4 Open-Drain (D0). Port 4 can be configured as an open-drain by resetting this bit (D0 = 0) or configured as push-pull active by setting this bit (D0 = 1). The default value is 1.

Port 5 Open-Drain (D4). Port 5 can be configured as an open-drain by resetting this bit (D4 = 0) or configured as push-pull active by setting this bit (D4 = 1). The default value is 1.

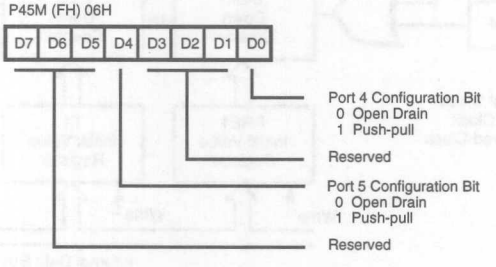


Figure 19. Port 4 and 5 Configuration Register (F) 06H [Write Only]

Power-On Reset (POR). A timer circuit clocked by a dedicated on-board RC oscillator is used for the Power-On Reset (POR) timer function. The POR time allows V_{CC} and the oscillator circuit to stabilize before instruction execution begins.

The POR timer circuit is a one-shot timer triggered by one of three conditions:

1. Power fail to Power OK status.
2. Stop-Mode Recovery (if D5 of SMR=1).
3. WDT timeout.

The POR time is a nominal 5 ms. Bit 5 of the STOP mode Register determines whether the POR timer is bypassed after Stop-Mode Recovery (typical for external clock, RC/LC oscillators).

HALT. HALT turns off the internal CPU clock, but not the XTAL oscillation. The counter/timers and external interrupts IRQ0, IRQ1, IRQ2, and IRQ3 remain active. The devices are recovered by interrupts, either externally or internally generated.

STOP. This instruction turns off the internal clock and external crystal oscillation. It reduces the standby current to 10 µA or less. The STOP mode is terminated by a reset only, either by WDT timeout, POR, SMR, or external reset. This causes the processor to restart the application program at address 000CH. In order to enter STOP (or HALT) mode, it is necessary to first flush the instruction pipeline to avoid suspending execution in mid-instruction. To do this, the user must execute a NOP (opcode=FFH) immediately before the appropriate sleep instruction, i.e.,

```
FF  NOP    ; clear the pipeline
6F  STOP   ; enter STOP mode
           or
FF  NOP    ; clear the pipeline
7F  HALT   ; enter HALT mode
```

Stop-Mode Recovery Register (SMR). This register selects the clock divide value and determines the mode of Stop-Mode Recovery (Figure 20). All bits are write only, except bit 7 which is read only. Bit 7 is a flag bit that is hardware set on the condition of STOP recovery and reset by a power-on cycle. Bit 6 controls whether a low level or a high level is required from the recovery source. Bit 5 controls the reset delay after recovery. Bits 2, 3, and 4, or the SMR register, specify the source of the Stop-Mode Recovery signal. Bits 0 and 1 determine the timeout period of the WDT. The SMR is located in Bank F of the Expanded Register group at address 0BH.

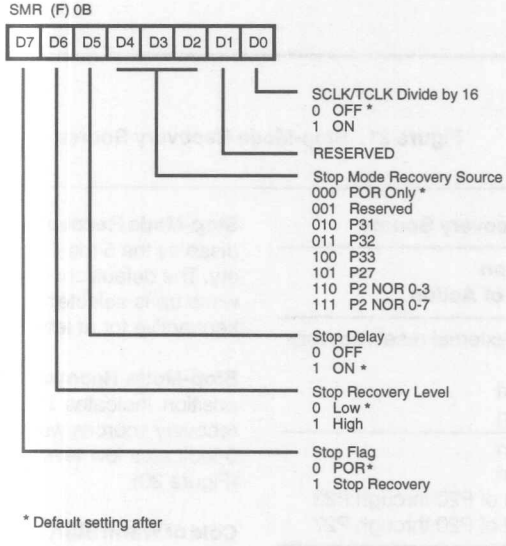


Figure 20. Stop-Mode Recovery Register (SMR)

Z8 FUNCTIONAL DESCRIPTION (Continued)

SCLK/TCLK Divide-By-16 Select (D0). D0 of the SMR controls a divide-by-16 prescaler of SCLK/TCLK. The purpose of this control is to selectively reduce device power consumption during normal processor execution (SCLK control) and/or HALT mode (where TCLK sources counter/timers and interrupt logic).

Stop-Mode Recovery Source (D4-D2). These three bits of the SMR specify the wake-up source of the STOP recovery (Figure 21 and Table 6).

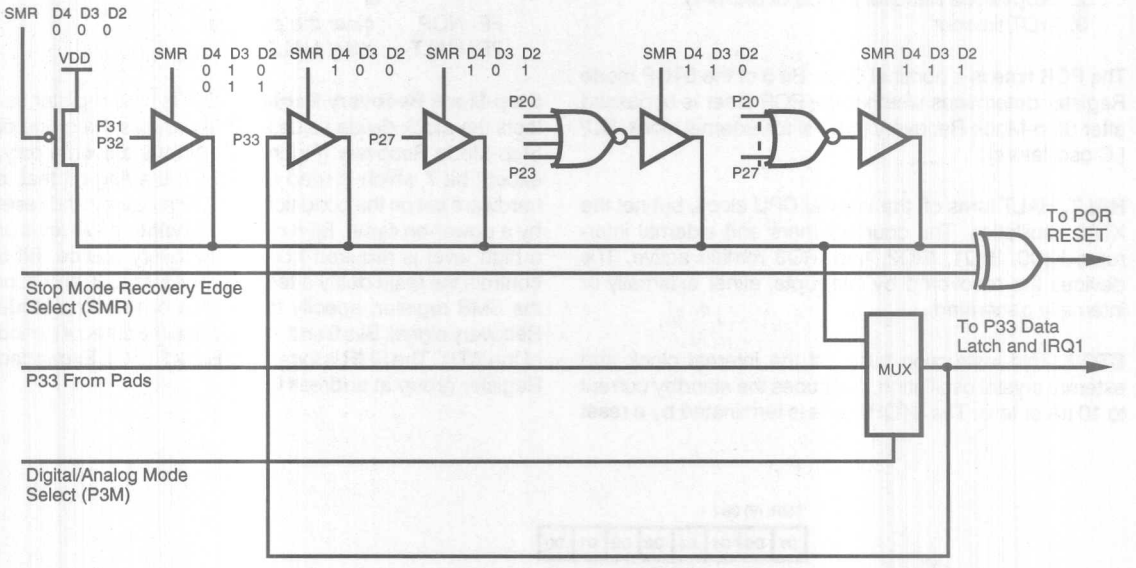


Figure 21. Stop-Mode Recovery Source

Table 6. Stop-Mode Recovery Source

SMR:432			Operation
D4	D3	D2	Description of Action
0	0	0	POR and/or external reset recovery
0	0	1	Reserved
0	1	0	P31 transition
0	1	1	P32 transition
1	0	0	P33 transition
1	0	1	P27 transition
1	1	0	Logical NOR of P20 through P23
1	1	1	Logical NOR of P20 through P27

Stop-Mode Recovery Delay Select (D5). This bit, if High, disables the 5 ms /RESET delay after Stop-Mode Recovery. The default configuration of this bit is one. If the "fast" wake up is selected, the Stop-Mode Recovery source is kept active for at least 5 TpC.

Stop-Mode Recovery Edge Select (D6). A 1 in this bit position indicates that a high level on any one of the recovery sources wakes the Z89120 from STOP mode. A 0 indicates low level recovery. The default is 0 on POR (Figure 20).

Cold or Warm Start (D7). This bit is set by the device upon entering STOP mode. It is active High, and is 0 (cold) on POR/WDT /RESET. This bit is read only. It is used to distinguish between cold or warm start.

DSP Control Register (DSPCON). The DSPCON register controls various aspects of the Z8 and the DSP. It can configure the internal system clock (SCLK) or the Z8, /RESET and HALT of the DSP, and control the interrupt interface between the Z8 and the DSP (Figure 22).

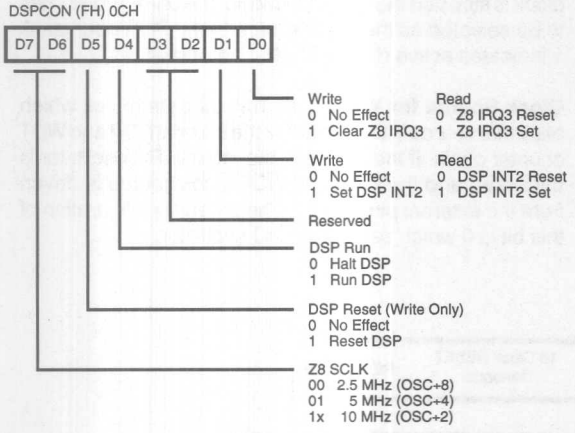


Figure 22. DSP Control Register (F) 0CH [Read/Write]

Z8 IRQ3 (D0). This bit, when read, indicates the status of Z8 IRQ3. Z8 IRQ3 is set by the DSP by writing to DSP Expanded Register 4. By writing a 1 to this bit, Z8 IRQ3 is /RESET.

DSP INT2 (D1). This bit is linked to DSP INT2. Writing a 1 to this bit sets DSP INT2. Reading this bit indicates the status of DSP INT2.

DSP RUN (D4). This bit defines the HALT mode of the DSP. If this bit is set to 0, then the DSP clock is turned off to minimize power consumption. After this bit is set to 1, then the DSP will continue code execution from where it was halted. After a hardware reset, this bit is set to 0.

DSP RESET (D5). Setting this bit to 1 will reset the DSP. If the DSP was in HALT mode, this bit is automatically preset to 1. Writing a 0 has no effect. This bit is write only.

Z8 SLCK (D8-D7). These bits define the SCLK frequency of the Z8. The oscillator can be either divided by 8, 4, or 2. After /RESET, both of these are defaulted to 00.

Watch-Dog Timer Mode Register (WDT). The WDT is a retriggerable one-shot timer that resets the Z8 if it reaches its terminal count. The WDT is initially enabled by executing the WDT instruction and refreshed on subsequent executions of the WDT instruction. The WDT circuit is driven by an on-board RC oscillator or external oscillator from the XTAL1 pin. The POR clock source is selected with bit 4 of the WDT register (Figure 23).

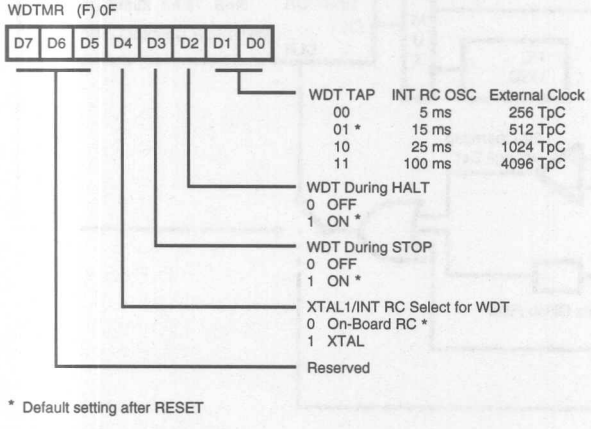


Figure 23. Watch-Dog Timer Mode Register

Z8 FUNCTIONAL DESCRIPTION (Continued)

WDT Time Select (D0, D1). Selects the WDT time period (Figure 24). It is configured as shown in Table 7.

Table 7. WDT Time Select

D1	D0	Timeout of Internal RC OSC	Timeout of XTAL clock
0	0	5 ms min	256 TpC
0	1	15 ms min	512 TpC
1	0	25 ms min	1024 TpC
1	1	100 ms min	4096 TpC

Notes:
TpC = XTAL clock cycle
The default on reset is 15 ms.
See Figures 54 to 57 for details.

WDT During HALT (D2). This bit determines whether or not the WDT is active during HALT mode. A 1 indicates active during HALT. The default is 1.

WDT During STOP (D3). This bit determines whether or not the WDT is active during STOP mode. Since XTAL clock is stopped during STOP mode, the on-board RC has to be selected as the clock source to the POR counter. A 1 indicates active during STOP. The default is 1.

Clock Source for WDT (D4). This bit determines which oscillator source is used to clock the internal POR and WDT counter chain. If the bit is a 1, the internal RC oscillator is bypassed and the POR and WDT clock source is driven from the external pin, XTAL1. The default configuration of this bit is 0 which selects the RC oscillator.

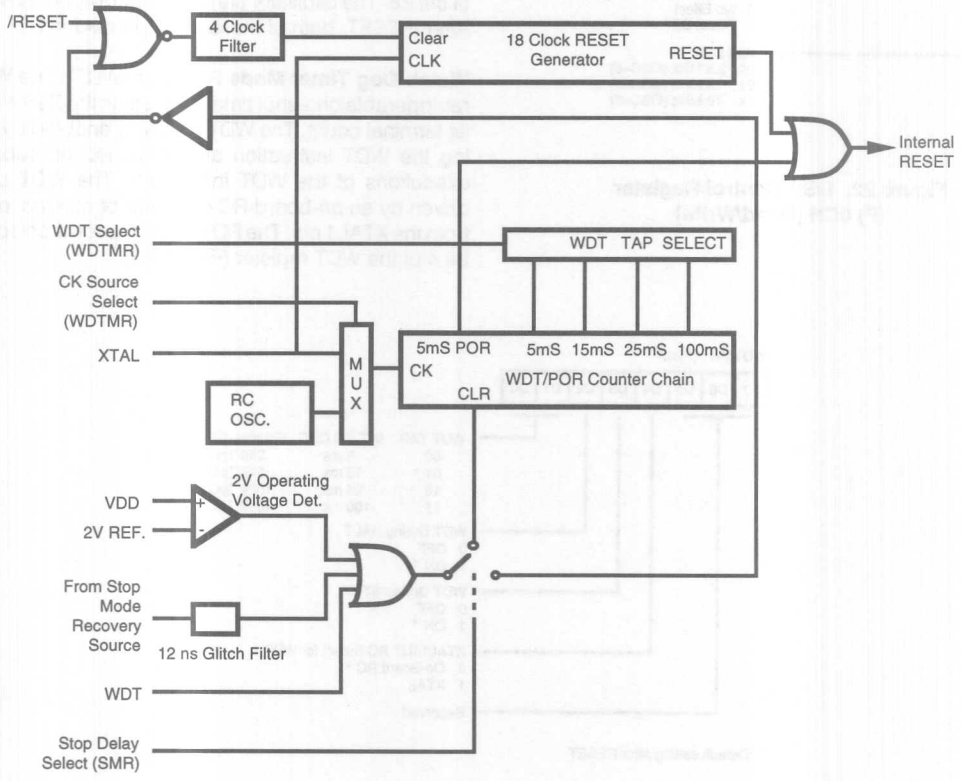


Figure 24. Resets and WDT

DSP FUNCTIONAL DESCRIPTION

General. The DSP is a high-performance second generation CMOS Digital Signal Processor with a modified Harvard-type architecture with separate program and data ports. The design has been optimized for processing power and saving silicon space.

Program Memory. Programs of up to 4K words can be masked into internal DSP ROM. Four locations are dedicated to the vector address for the three interrupts (0FFDH-0FFFH) and the starting address following a reset (0FFCH).

Internal Data RAM. The DSP has an internal 512 x 16-bit word data RAM organized as two banks of 256 x 16-bit words each, referred to as RAM0 and RAM1. Each data RAM bank is addressed by three address register pointers, referred to as P0:0-P2:0 for RAM0 and P0:1-P2:1 for RAM1. Three addressing modes are available to access the data RAM: register indirect, direct, and short-form direct addressing. These modes are discussed in detail in Functional Description. The contents of each RAM can be loaded simultaneously into the X and Y inputs of the multiplier.

Registers. The DSP has twelve internal registers and seven extended registers. The extended registers are for the A/D and D/A converters, and the mailbox and interrupt interfacing between DSP to the Z8. Extended registers are accessed in one machine cycle, the same as internal registers.

Instruction Timing. Many instructions are executed in one machine cycle. Long immediate instructions and Branch or Call instructions are executed in two machine cycles. When the program memory is referenced in internal RAM indirect mode, it takes three machine cycles. In addition, one more machine cycle is required if the PC is selected as the destination of a data transfer instruction. This only happens in the case of a register indirect branch instruction.

For example, an a(i) * b(j) + Acc → Acc calculation is done in one machine cycle, modifying the RAM pointer contents. Both operands, a(i) and b(j), can be located in two independent RAM (0 and 1) addresses.

Multiply/Accumulate. The multiplier can perform a 16-bit x 16-bit multiply or multiply accumulate in one machine cycle using the Accumulator and/or both the X and Y inputs. The multiplier produces a 32-bit result, however, only the 24 most significant bits are saved for the next instruction or accumulation. For operations on very small numbers where the least significant bits are important, the data should first be scaled by eight bits to avoid truncation errors.

ALU. The 24-bit ALU has two input ports, one of which is connected to the output of the 24-bit Accumulator. The other input is connected to the 24-bit P-Bus, the upper 16 bits of which are connected to the 16-bit D-Bus. A shifter between the P-Bus and the ALU input port can shift the data by three bits right, during a multiply/accumulator operation or no shift.

Hardware Stack. A six-level hardware stack is connected to the D-Bus to hold subroutine return addresses or data. The CALL instruction pushes PC+2 onto the stack. The RET instruction pops the contents of the stack to the PC.

Interrupts. The DSP has three positive edge triggered interrupt inputs. An interrupt is acknowledged at the end of any instruction execution. It takes two machine cycles to enter an interrupt instruction sequence. The PC is pushed onto the stack. A RET instruction transfers the contents of the stack to the PC and decrements the stack pointer by one word. The priority of the interrupts is 0 = highest, 2 = lowest.

DSP Registers

There are 15 internal and extended 16-bit registers which are defined in Table 8.

Table 8. DSP Registers

Register	Attribute	Register Definition
BUS	Read	Data-Bus
X	Read/Write	X Multiplier Input, 16-Bit
Y	Read/Write	Y Multiplier Input, 16-Bit
A	Read/Write	Accumulator, 24-Bit
SR	Read/Write	Status Register
SP	Read/Write	Stack Pointer
PC	Read/Write	Program Counter
P	Read	Output of MAC, 24-Bit
EXT0	Read	Z8 ERF Bank B, Register 00-01 (from Z8)
	Write	Z8 ERF Bank B, Register 08-09 (to Z8)
EXT1	Read	Z8 ERF Bank B, Register 02-03 (from Z8)
	Write	Z8 ERF Bank B, Register 0A-0B (to Z8)
EXT2	Read	Z8 ERF Bank B, Register 04-05 (from Z8)
	Write	Z8 ERF Bank B, Register 0C-0D (to Z8)
EXT3	Read	Z8 ERF Bank B, Register 06-07 (from Z8)
	Write	Z8 ERF Bank B, Register 0E-0F (to Z8)
EXT4	Read/Write	DSP Interrupt Control Register
EXT5	Read	A/D Converter
	Write	D/A Converter
EXT6	Read/Write	Analog Control Register

Two registers, Bus and P are read only. If either of these registers are designated as the destination of a data transfer instruction, the contents will be unaffected.

BUS is a read-only register which, when accessed, returns the contents of the D-Bus.

X and **Y** are two 16-bit input registers for the multiplier. These registers can be utilized as temporary registers when the multiplier is not being used. The P register is affected by changing X or Y.

A is a 24-bit Accumulator. The output of the ALU is sent to this register. When 16-bit data is transferred into this register, it goes into the 16 MSB's and the least significant eight bits are set to zero. Only the upper 16 bits are transferred to the destination register when the Accumulator is selected as a source register in transfer instructions.

SR is the DSP status register (Figure 25) which contains the ALU status and certain control bits as shown in Table 9.

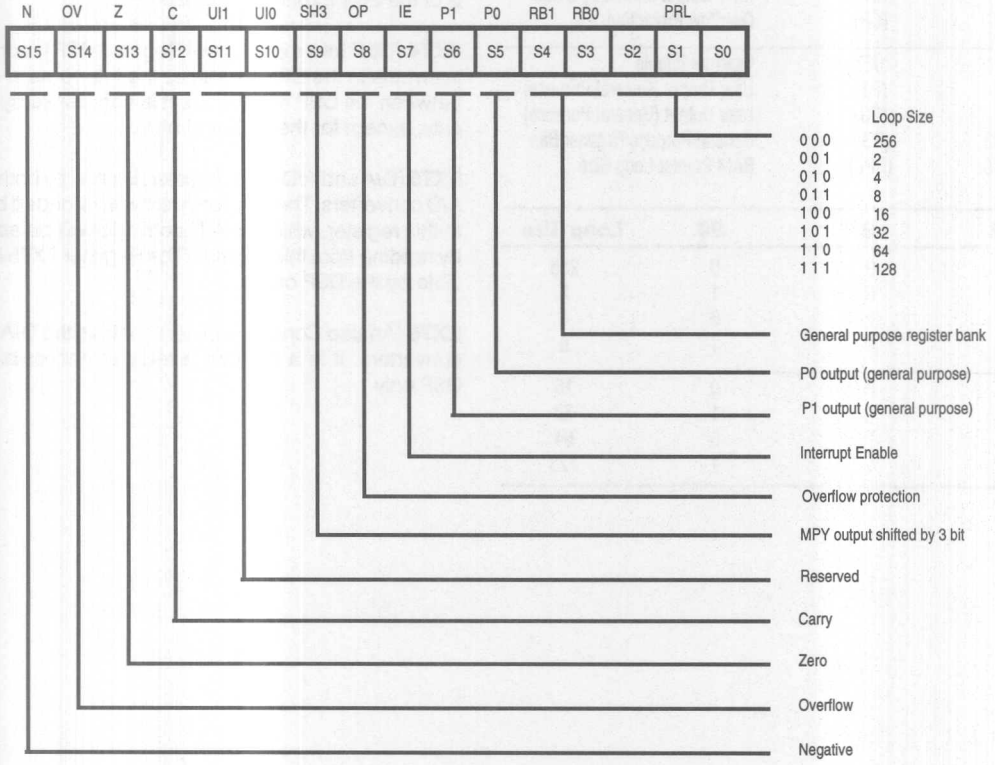


Figure 25. DSP Status Register

DSP FUNCTIONAL DESCRIPTION (Continued)

Table 9. DSP Status Register Bits

Status	Register Bit	Function
S15	(N)	ALU Negative
S14	(OV)	ALU Overflow
S13	(Z)	ALU Zero
S12	(C)	Carry
S11	(UO1)	User Pin 1 Input (DSP1)
S10	(UO0)	User Pin 0 Input (DSP0)
S9	(SH3)	MPY Output Shifted by 3 Bits
S8	(OP)	Overflow Protection
S7	(IE)	Interrupt Enable
S6	(P1)	User Output (General Purpose)
S5	(P0)	User Output (General Purpose)
S4-S3	(RBi)	General Purpose Register Ban
S2-S0	(RPL)	RAM Pointer Loop Size

S2	S1	S0	Loop Size
0	0	0	256
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

PC is the program counter. When this register is assigned as a destination register, one NOP machine cycle is added automatically to adjust the pipeline timing.

P is the output register for the 24-bit multiplier.

EXT3-EXT0 (Extended Registers 0-3) are the Mailbox Registers in which the DSP and the Z8 communicate (Figure 26). These four 16 bit registers correspond to the eight outgoing and eight incoming 8-bit registers in Bank B of the Z8's Expanded Register File.

EXT4 (DSP Interrupt Control Register (ICR)) controls the interrupts in the DSP as well as the interrupts in common between the DSP and the Z8. It is accessible by the DSP only, except for the bit F and bit 9.

EXT5 (D/A and A/D Data Register) is used by both D/A and A/D converters. The D/A converter will be loaded by writing to this register, while the A/D converter will be addressed by reading from this register. The Register EXT5 is accessible by the DSP only.

EXT6 (Analog Control Register) controls the D/A and A/D converters. It is a read/write register accessible by the DSP only.

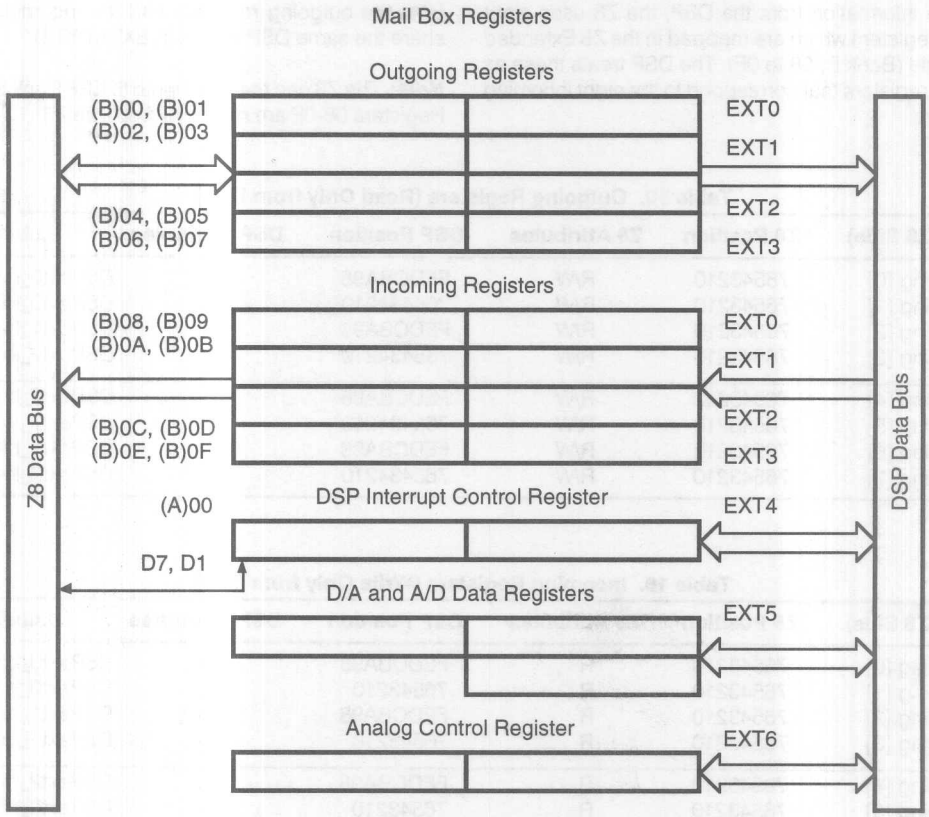


Figure 26. Z8-DSP Interface

3

DSP-Z8 Mailbox

To receive information from the DSP, the Z8 uses eight incoming registers which are mapped in the Z8 Extended Register File (Bank B, 08 to 0F). The DSP treats these as four 16-bit registers that correspond to the eight incoming Z8 registers.

Both the outgoing registers and the incoming registers share the same DSP address (EXT3-EXT0).

Note: The Z8 can read and write to ERF Bank B R00-R07, Registers 08-0F are read only from the Z8.

Table 10. Outgoing Registers (Read Only from DSP)

Field (Z8 Side)	Z8 Position	Z8 Attributes	DSP Position	DSP Attributes	Label
Outgoing [0]	76543210	R/W	FEDCBA98	R	DSPext0_hi (15-8)
Outgoing [1]	76543210	R/W	765434210	R	DSPext0_lo (7-0)
Outgoing [2]	76543210	R/W	FEDCBA98	R	DSPext1_hi (15-8)
Outgoing [3]	76543210	R/W	765434210	R	DSPext1_lo (7-0)
Outgoing [4]	76543210	R/W	FEDCBA98	R	DSPext2_hi (15-8)
Outgoing [5]	76543210	R/W	765434210	R	DSPext2_lo (7-0)
Outgoing [6]	76543210	R/W	FEDCBA98	R	DSPext3_hi (15-8)
Outgoing [7]	76543210	R/W	765434210	R	DSPext3_lo (7-0)

Table 10. Incoming Registers (Write Only from DSP)

Field (Z8 Side)	Z8 Position	Z8 Attributes	DSP Position	DSP Attributes	Label
Incoming [0]	76543210	R	FEDCBA98	W	DSPext0_hi (15-8)
Incoming [1]	76543210	R	76543210	W	DSPext0_lo (7-0)
Incoming [2]	76543210	R	FEDCBA98	W	DSPext1_hi (15-8)
Incoming [3]	76543210	R	76543210	W	DSPext1_lo (7-0)
Incoming [4]	76543210	R	FEDCBA98	W	DSPext2_hi (15-8)
Incoming [5]	76543210	R	76543210	W	DSPext2_lo (7-0)
Incoming [6]	76543210	R	FEDCBA98	W	DSPext3_hi (15-8)
Incoming [7]	76543210	R	76543210	W	DSPext3_lo (7-0)

DSP Interrupts

The DSP processor has three interrupt sources (INT2, INT1, INT0) (Figure 27). These sources have different priority levels (Figure 28). The highest priority, the next lower and the lowest priority level are assigned to INT2, INT1, and INT0, respectively. The DSP does not allow interrupt nesting (interrupting service routines that are currently being executed). When two interrupt requests occur simultaneously, the DSP starts servicing the interrupt with the highest priority level (Figure 29).

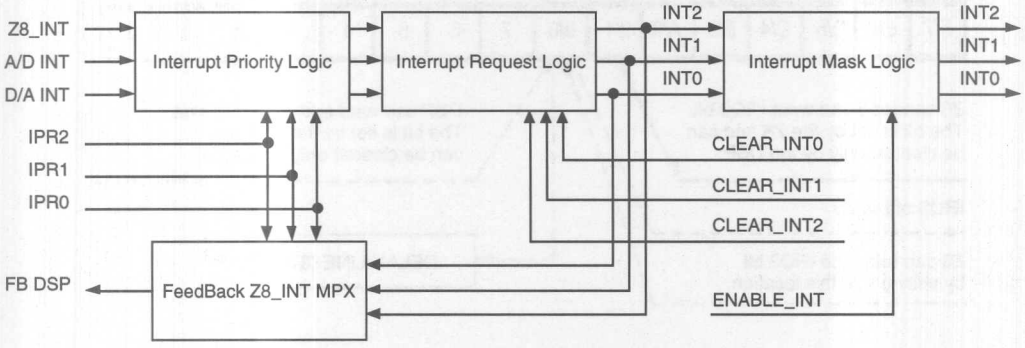


Figure 27. DSP Interrupts

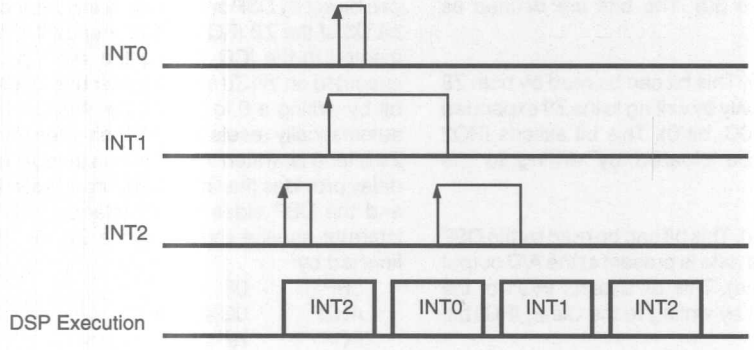


Figure 28. DSP Interrupt Priority Structure

DSP Interrupts (Continued)

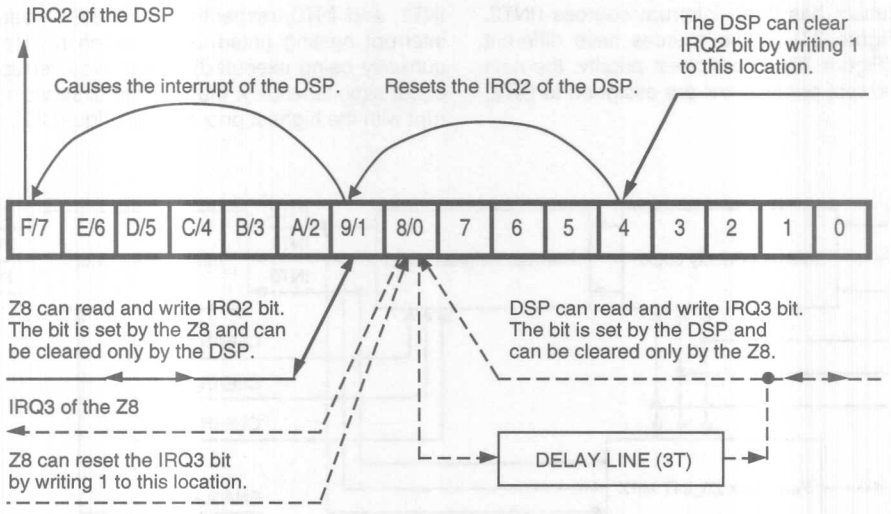


Figure 29. Interprocessor Interrupts Structure

Interrupt Control Register (ICR). The ICR is mapped into EXT4 of the DSP (Figure 30). The bits are defined as follows.

DSP_IRQ2 (Z8 Interrupt). This bit can be read by both Z8 and DSP and can be set only by writing to the Z8 expanded Register File (Bank F, R0C, bit 0). This bit asserts IRQ2 of the DSP and can be cleared by writing to the Clear_IRQ2 bit.

DSP_IRQ1 (A/D Interrupt). This bit can be read by the DSP only and is set when valid data is present at the A/D output register (conversion done). This bit asserts IRQ1 of the DSP and can be cleared by writing to the Clear_IRQ1bit.

DSP_IRQ0 (D/A Interrupt). This bit can be read by DSP only and is set by Timer3. This bit assists IRQ0 of the DSP and can be cleared by writing to the Clear_IRQ0 bit.

DSP_MaskIntX. These bits can be accessed by the DSP only. Writing a 1 to these locations allows the INT to be serviced, while writing a 0 masks the corresponding INT off.

Z8_IRQ3. This bit can be read from both Z8 and DSP and can be set by DSP only. Addressing this location accesses bit D3 of the Z8 IRQ register, hence this bit is not implemented in the ICR. During the interrupt service routine executed on the Z8 side, the user has to reset the Z8_IRQ3 bit by writing a 0 to bit D9. The hardware of the Z89120 automatically resets Z8_IRQ3 bit three instructions of the Z8 after 0 is written to its location in register bank 0F. This delay provides the timing synchronization between the Z8 and the DSP sides during interrupts. In summary, the interrupt service routine of the Z8 for IRQ3 should be finished by:

```
SRP     0F
AND     0C,#%FD
POP     RP
IRET
```

DSP Enable_INT. Writing a 1 to this location enables global interrupts of the DSP while writing 0 disables them. A system reset globally disables all interrupts.

DSP_IPRX. This 3-bit group defines the Interrupt Priority according to Table 12.

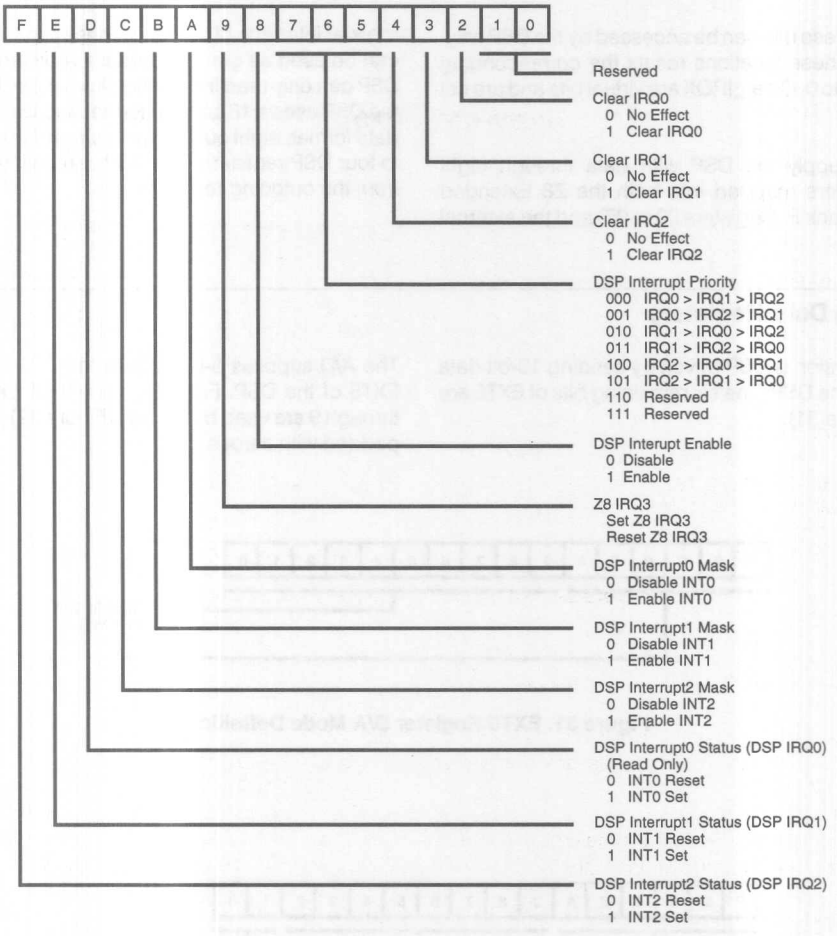


Figure 30. EXT4 DSP Interrupt Control Register (ICR) Definition

Table 12. DSP Interrupt Priority

High Priority int0 Interrupt	Medium Priority int1 Interrupt	Low Priority int2 Interrupt	DSP_IPR2, 1, 0
IRQ0	IRQ1	IRQ2	0 0 0
IRQ0	IRQ2	IRQ1	0 0 1
IRQ1	IRQ0	IRQ2	0 1 0
IRQ1	IRQ2	IRQ0	0 1 1
IRQ2	IRQ0	IRQ1	1 0 0
IRQ2	IRQ1	IRQ0	1 0 1

DSP Interrupts (Continued)

Clear_IRQX. These bits can be accessed by the DSP only. Writing a 1 to these locations resets the corresponding DSP_IRQX bits to 0. Clear_IRQX are virtual bits and are not implemented.

The Z8 can supply the DSP with data through eight outgoing registers mapped into both the Z8 Extended Register File (Bank B, Registers 00 to 07) and the external register interface of the DSP. These registers are R/W and can be used as general purpose registers of the Z8. The DSP can only read information from these registers. Since the DSP uses a 16-bit data format and the Z8 uses an 8-bit data format, eight outgoing registers of the Z8 correspond to four DSP registers. The DSP can only read information from the outgoing registers.

DSP Analog Data Registers

The D/A conversion is DSP driven by sending 10-bit data to the EXT5 of the DSP. The six remaining bits of EXT5 are not used (Figure 31).

The A/D supplies 8-bit data to the DSP through register EXT5 of the DSP. From the 16 bits of EXT5, only bits 2 through 9 are used by the A/D (Figure 32). Bits 0 and 1 are padded with zeroes.

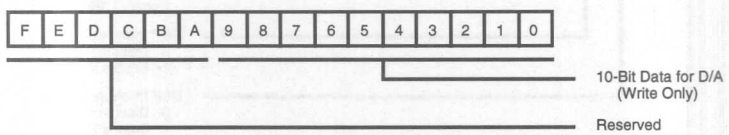


Figure 31. EXT5 Register D/A Mode Definition

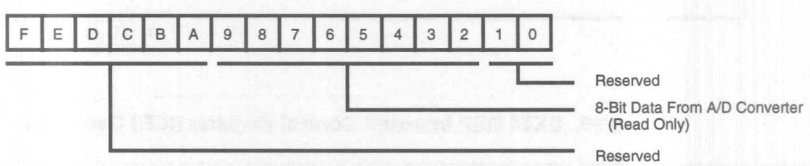


Figure 32. EXT5 Register A/D Mode Definition

Analog Control Register (ACR)

The Analog Control Register is mapped to register EXT6 of the DSP (Figure 33). This read/write register is accessible by the DSP only.

The 16-bit field of EXT6 defines modes of both the A/D and the D/A. The High Byte configures the D/A, while the Low Byte controls the A/D mode.

DSP IRQ0. Defines the source of DSP IRQ0 interrupt.

D/A Converter Effective Sampling Rate. This field defines the effective sampling rate of the D/A output (Figure 33). It changes the period of Timer3, which generates the interrupt for updating the output sample and in turn affects the maximum possible accuracy of the D/A (Table 13).

Table 13. D/A Data Accuracy

Sampling Rate	D/A Accuracy
64 kHz	8 Bits
16 kHz	10 Bits
10 kHz	10 Bits
4 kHz	10 Bits

DSP0. DSP0 is a general purpose output pin connected to bit 6. This bit has no special significance and may be used to output data by writing to bit 6.

DSP1. DSP1 is a general purpose output pin connected to bit 7. This bit has no special significance and may be used to output data by writing to bit 7.

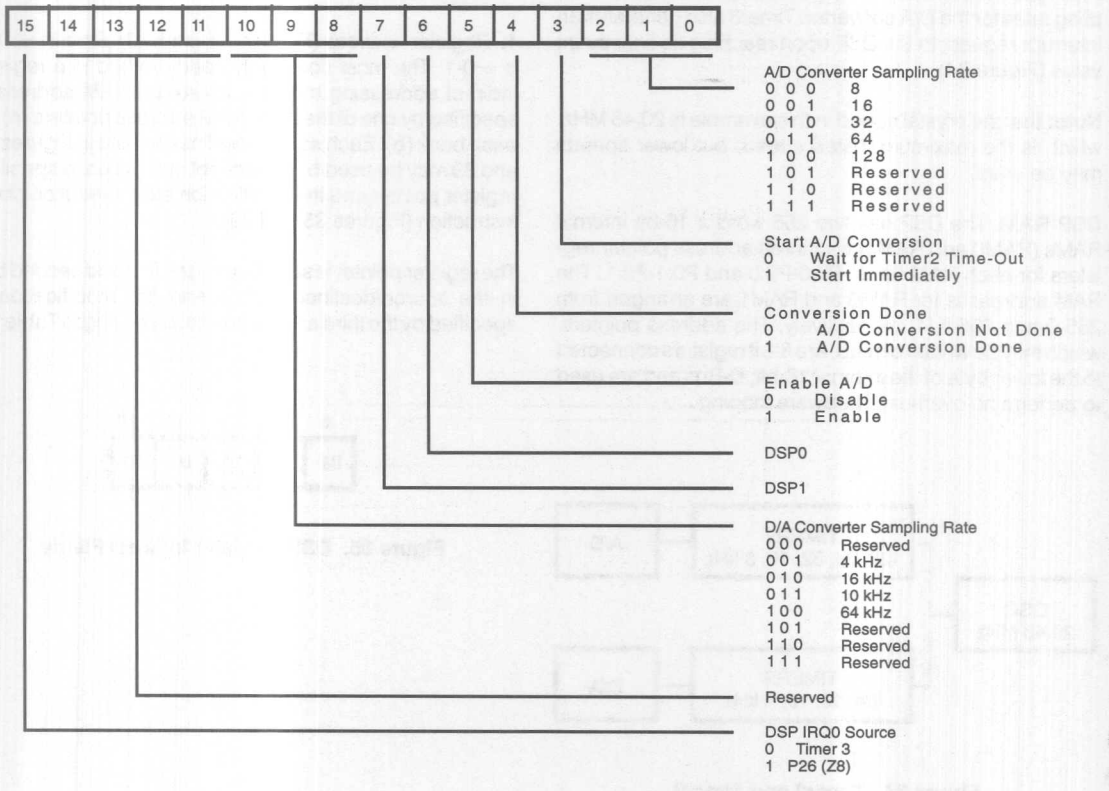


Figure 33. EXT6 Analog Control Register (ACR)

Analog Control Register (Continued)

Enable A/D. Writing a 0 to this location disables the A/D converter, a 1 will enable it. A hardware reset forces this bit to be 0.

Conversion Done. This read only flag indicates that the A/D conversion is complete. Upon reading EXT5 (A/D data), the Conversion Done flag is cleared.

Start A/D Conversion. Writing a 1 to this location immediately starts one conversion cycle. If this bit is reset to 0 the input data is converted upon successive Timer2 time-outs. A hardware reset forces this bit to be 1.

A/D Converter Sampling Rate. This field defines the sampling rate of the A/D. It changes the period of Timer2 interrupt (Figure 33).

DSP Timers

Timer2 is a free running counter that divides the XTAL frequency to support different sampling rates for the A/D converter. The sampling rate is defined by the Analog Control Register. Upon reaching the end of a count, the timer generates an interrupt request to the DSP.

Analogous to Timer2, Timer3 generates the different sampling rates for the D/A converter. Timer3 also generates an interrupt request to the DSP upon reaching its final count value (Figure 34).

Note: that the crystal speed in this example is 20.48 MHz, which is the maximum tested speed, but lower speeds may be used.

DSP RAM. The DSP has two 256 word x 16-bit internal RAMs (RAM0 and RAM1) with three address pointer registers for each RAM Bank, P0:0-P2:0 and P0:1-P2:1. The RAM addresses for RAM0 and RAM1 are arranged from 255-0 and 256-511, respectively. The address pointers, which may be written or read, are 8-bit registers connected to the lower byte of the internal 16-bit, D-Bus and are used to perform no overhead hardware looping.

The contents of the RAM can be read or written in one machine cycle per word without disturbing any internal registers or status other than the RAM address pointer used for each RAM.

The address of the RAM is specified in one of three ways:

1. Register Indirect (Figures 35 and 38) Pn:b n = 0-2, b = 0-1: The most commonly used method is a register indirect addressing method, where the RAM address is specified by one of the three RAM address pointers (n) for each bank (b). Each source/destination field in Figures 35 and 39 may be used by an indirect instruction to specify a register pointer and its modification after execution of the instruction (Figures 35 and 39).

The register pointer is specified by the first and second bits in the source/destination field and the modification is specified by the third and fourth bits according to Table 14.

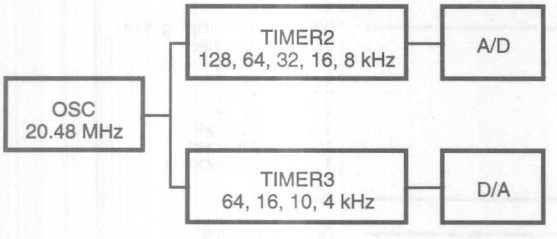


Figure 34. Timer2 and Timer3

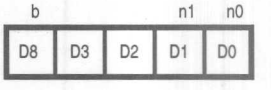


Figure 35. DSP Register Indirect Fields

Table 14. Register Indirect Fields

S/D Field	Modification	Meaning
00xx	NOP	No Operation
01xx	+1	Simple increment
10xx	–1/LOOP	Decrement modulo the loop count
11xx	+1/LOOP	Increment modulo the loop count
xx00	P0:0 or P0:1*	See note a
xx01	P1:0 or P1:1*	See note a
xx10	P2:0 or P2:1*	See note a

Notes:
a If bit 8 is zero, P0:0-2:0 are selected; if bit 8 is one, P0:1-2:1 are selected.
* P0:0-P2:0 and P0:1-P2:1 refer to the DSP pointer registers and not to the I/O ports in the Z8.

When LOOP mode is selected, the size of the loop is obtained from the least-most-significant three bits of the Status Register. The increment or decrement of the register is accomplished modulo the loop size. As an example, if the loop size is specified as 32 by entering the value 101 into bits 2-0 of the Status Register (S2-S0) and an increment +1/LOOP is specified in the address field of the instruction, i.e., the RPi field is 11xx, then the register specified by RPi will increment, but only the least significant five bits will be affected.

2. Register Direct (Figure 36): The second method is a direct addressing method. The address of the RAM is specified by the address field of the instruction directly. Because this addressing method consumes nine bits (0-511) of the instruction field, some instructions cannot use this mode.

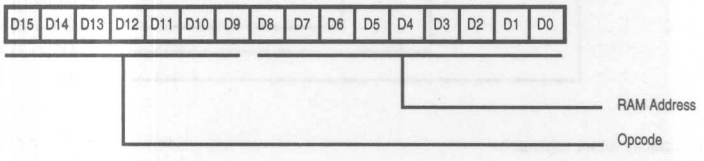


Figure 36. DSP Internal RAM Address Format

DSP Timers (Continued)

3. Short Form Direct (Figure 37) Dn:b n = 0-3, b = 0-1: The last method is called Short Form Direct Addressing, where one-out-of 32 addresses in internal RAM can be specified. The 32 addresses are the 16 Low addresses in RAM Bank0 and the 16 Low addresses in RAM Bank 1. Bit 8 of the instruction field determines RAM Bank 0 or 1. The 16 addresses are determined by a 4-bit code comprised of bits S3 and S4 of the status register and the third and fourth bits of the Source/Destination field. Because this mode can specify a direct address in a short form, all the instructions where the register indirect mode is used can use this mode.

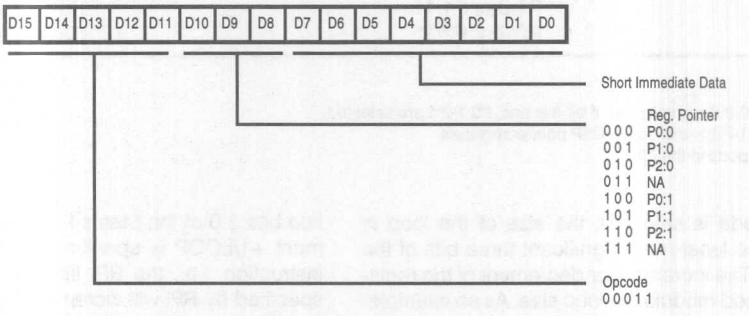
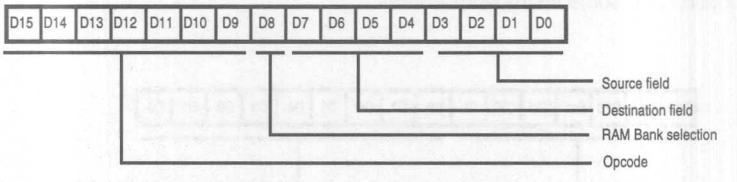


Figure 37. Short Form Direct Address

INSTRUCTION FORMAT



Note:
Source/Destination fields can specify either register or RAM addresses in RAM pointer indirect mode.

Figure 38. General Instruction Format

Table 15. Registers Fields

Source/Destination	Register
0000	BUS**
0001	X
0010	Y
0011	A
0100	S
0101	ST
0110	PC
0111	P**
1000	EXT0
1001	EXT1
1010	EXT2
1011	EXT3
1100	EXT4
1101	EXT5
1110	EXT6
1111	Reserved

Table 16. Register Pointers Fields

Source/Destination	Meaning
00xx	NOP
01xx	+1
10xx	–1/LOOP
11xx	+1/LOOP
xx00	P0:0 or P0:1*
xx01	P1:0 or P1:1*
xx10	P2:0 or P2:1*

Notes:

* If RAM Bank bit is 0 then P0:0-P2:0 are selected. If RAM Bank bit is 1 then P0:1-P2:1 are selected. Also note, P0:0-P2:0 and P0:1-P2:1 refer to the DSP pointer registers and not to the I/O ports in the Z8.

** Read only.

S4, S3 = bits 4, 3 of Status Register

D3, D2 = bits 4, 3 of Source/Destination Field

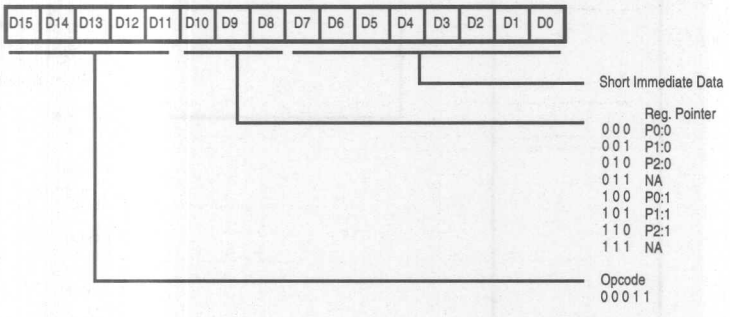


Figure 39. Short Immediate Data Load Format

3

INSTRUCTION FORMAT (Continued)

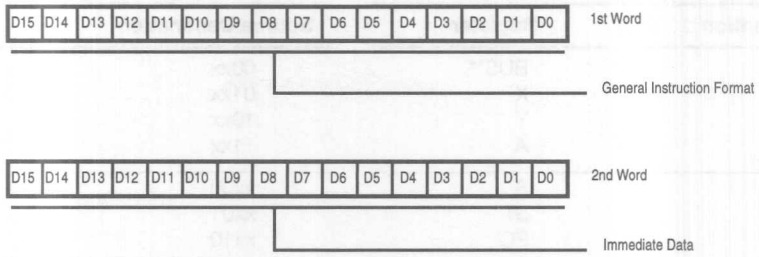


Figure 40. Immediate Data Load Format

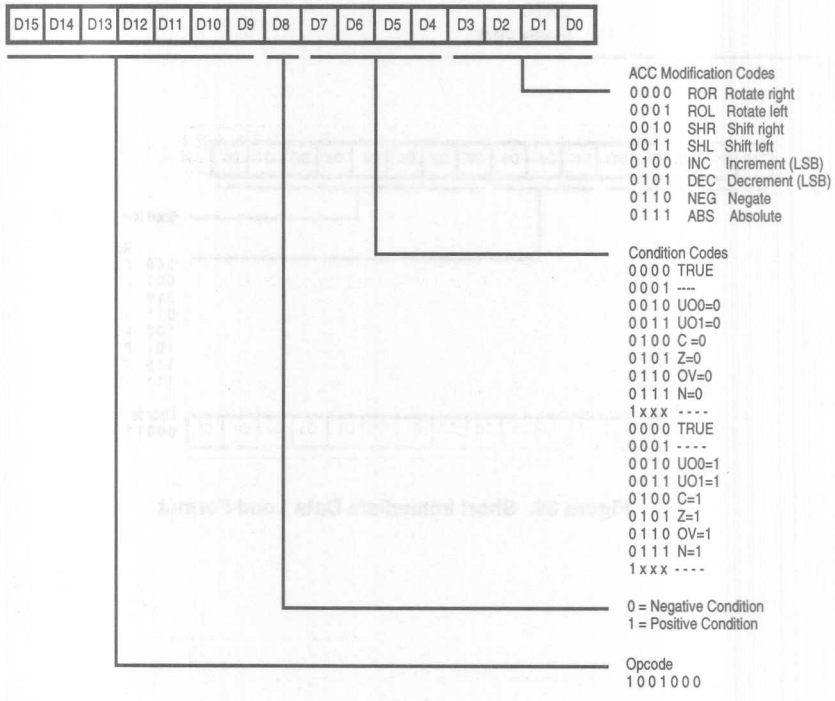


Figure 41. Accumulator Modification Format

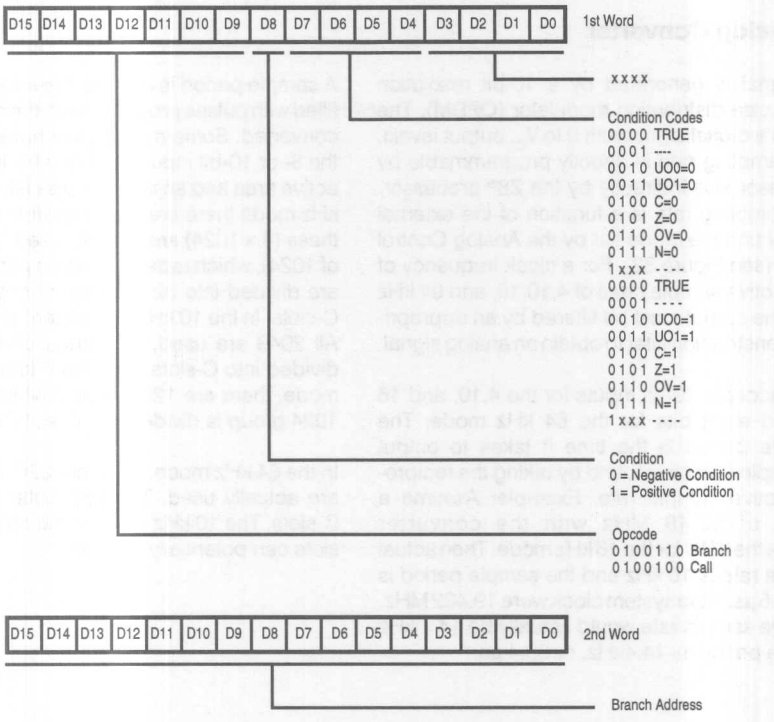


Figure 42. Branching Format

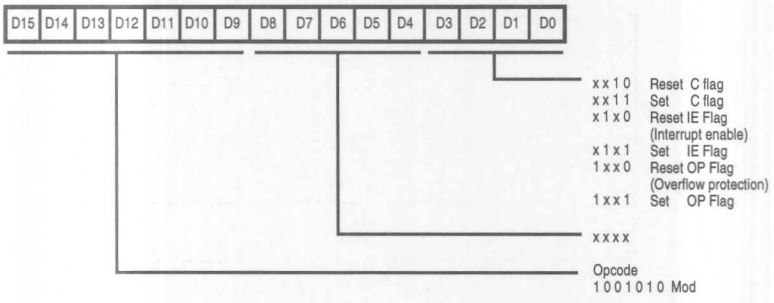


Figure 43. Flag Modification Format

PULSE WIDTH MODULATOR (PWM)

Digital to Analog Converter

The analog signal is generated by a 10-bit resolution oversampling pulse distribution modulator (OPDM). The OPDM output is a digital signal with 0 to V_{CC} output levels. The effective sampling rate is directly programmable by the DSP processor and indirectly by the Z8® processor. The effective sampling rate is a function of the external clock frequency and the mode set by the Analog Control Register (ACR) (see Figure 33). For a clock frequency of 20.48 MHz, effective sample rates of 4,10,16, and 64 kHz are available. The output must be filtered by an appropriate external reconstruction filter to obtain an analog signal.

The converter accepts 10-bit inputs for the 4,10, and 16 kHz modes and eight bits for the 64 kHz mode. The effective sample period is the time it takes to output successive samples which is found by taking the reciprocal of the effective sample rate. Example: Assume a system clock of 20.48 MHz with the converter programmed via the ADC for the 16 kHz mode. Then actual effective sample rate is 16 kHz and the sample period is $1 \div 16 \times 10^3$, or 62.5 µs. If the system clock were 19.432 MHz, then the effective sample rate would actually be 14.4 kHz and the sample period 1÷14.4kHz, or 69.4 µs.

A sample period is divided into small time slots which are filled with pulses proportional to the digital input value to be converted. Some modes may have more time slots than the 8- or 10-bit input value can fill. In this case there is an active area and an inactive area filled with zeroes. In the 4 kHz mode there are 5120 possible slots of which 4096 of these (4 x 1024) are actually used, (divided into 4 groups of 1024), which repeat the same pattern. These 1024 slots are divided into 32 equidistant groups of 32 bits called C-slots. In the 10 kHz mode there are 2048 possible slots. All 2048 are used, (2 groups of 1024) which are then divided into C-slots as in the 4 kHz mode. In the 16 kHz mode, there are 1280 slots of which 1024 are used. The 1024 group is divided into C-slots as in the 4 kHz mode.

In the 64 kHz mode, there are 320 time slots of which 256 are actually used. The 256 slots are divided into eight C-slots. The 10 kHz mode is the only mode where all time slots can potentially be filled.

Digital to Analog Converter (Continued)

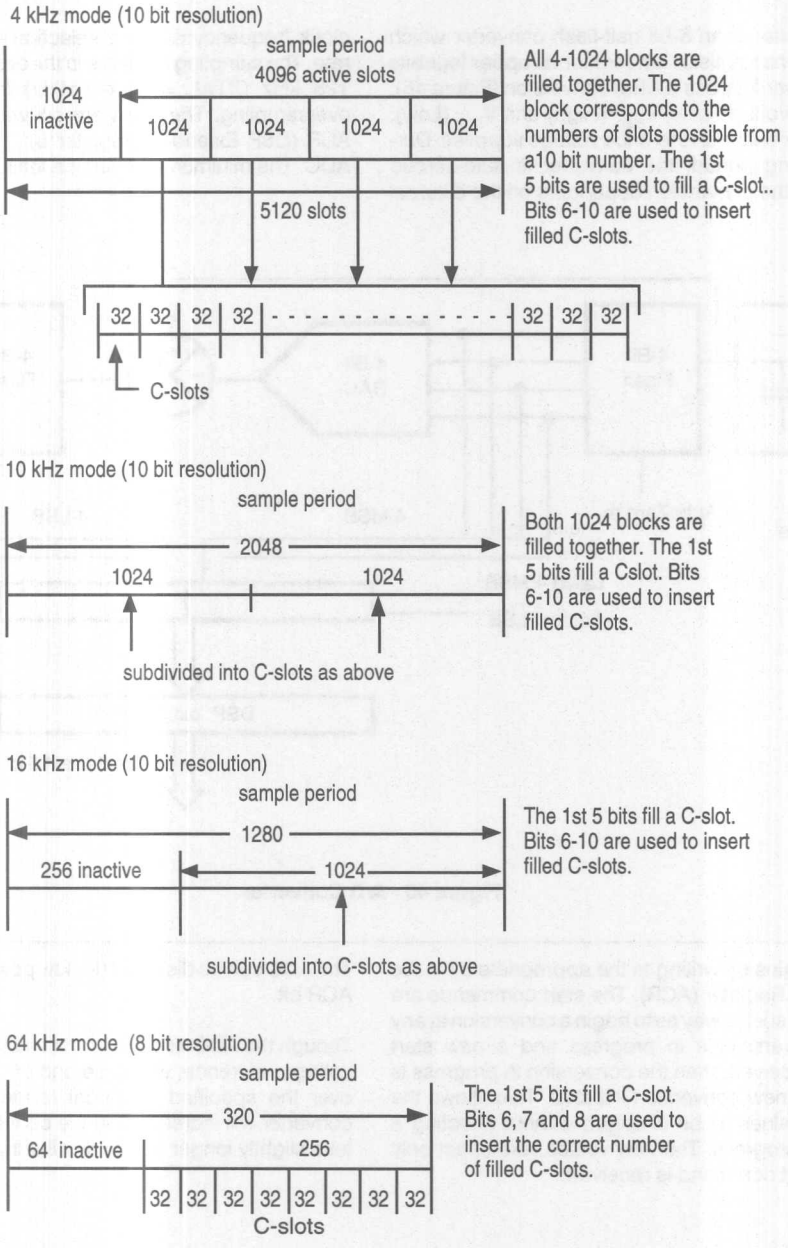


Figure 44. PWM Output

A/D CONVERTER (ADC)

Analog To Digital Converter

The A/D converter is an 8-bit half-flash converter which uses two reference resistor ladders for its upper four bits (MSBs) and lower four bits (LSBs) conversion (Figure 45). Two reference voltage pins, V_{REF+} (High) and V_{REF-} (Low), are provided for external reference voltage supplies. During the sampling period, the converter is auto-zeroed before starting the conversion depending on the external clock frequency and the selection of the A/D sampling rate. The sampling rates are in the order of 8, 16, 32, 64, or 128 kHz (XTAL = 20.48 MHz) in order to provide oversampling. The rates are software controlled by the ACR (DSP Extended Register 6). Timer2 supports the ADC. The minimum conversion time is 2 µs.

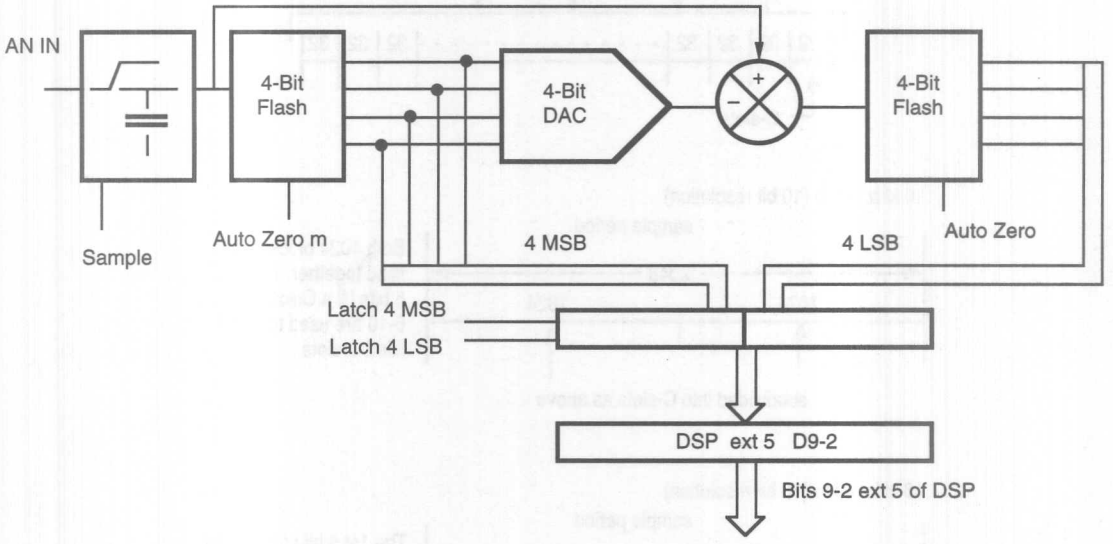


Figure 45. A/D Converter

Conversion begins by writing to the appropriate bit in the Analog Control Register (ACR). The start commands are implemented in such a way as to begin a conversion at any time. If a conversion is in progress and a new start command is received, then the conversion in progress is aborted and a new conversion initiated. This allows the programmed values to be changed without affecting a conversion-in-progress. The new values take effect only after a new start command is received.

The ADC can be disabled (for low power) or enabled by an ACR bit.

Though the ADC functions for a smaller input voltage and voltage reference, the noise and offsets remain constant over the specified electrical range. The errors of the converter will increase and the conversion time may also take slightly longer due to smaller input signals.

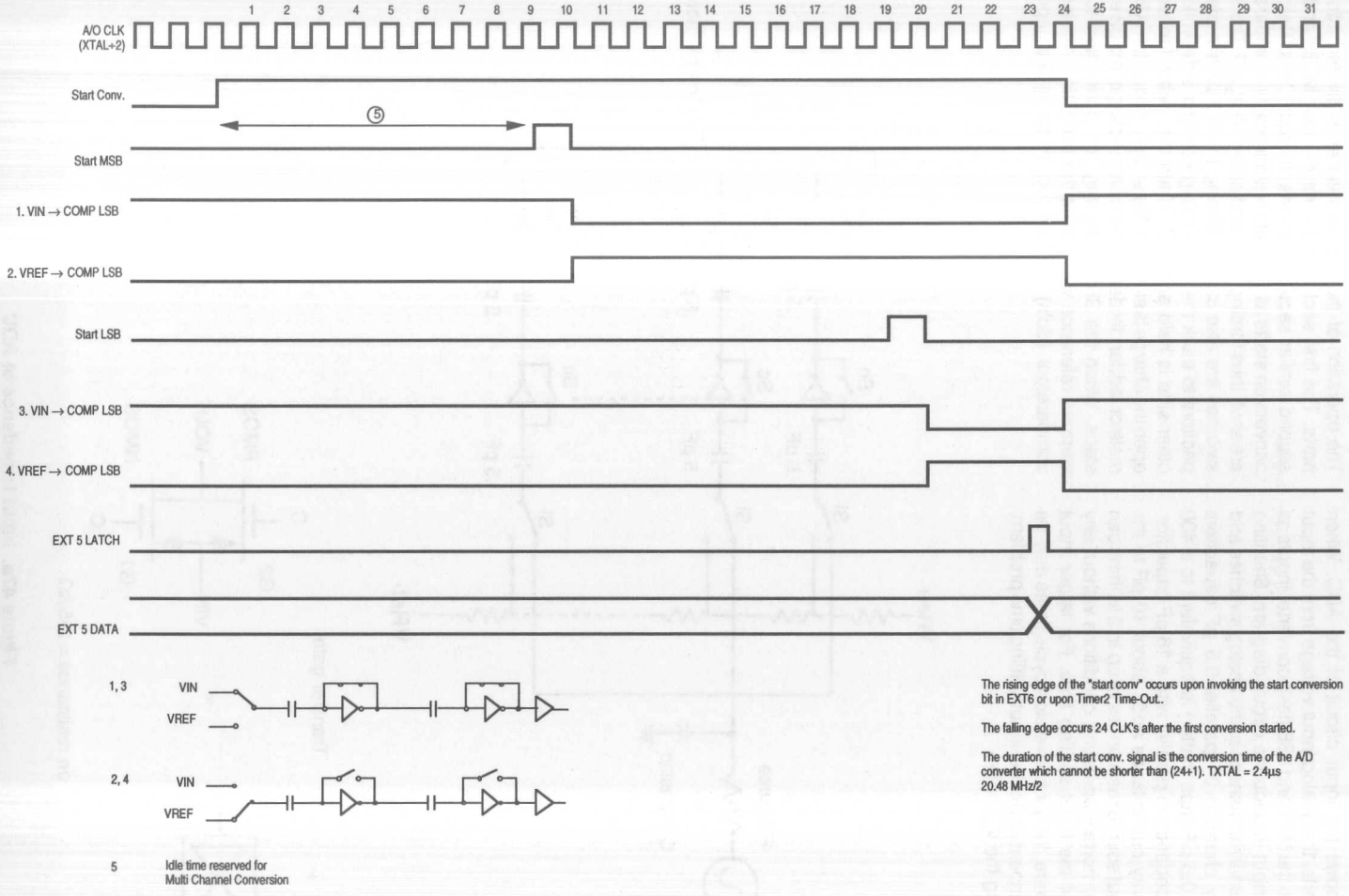


Figure 46. ADC Timing Diagram

A/D CONVERTER (ADC) (Continued)

Figure 47a shows the input circuit of the ADC. When conversion starts, the analog input voltage from the input is connected to the MSB and LSB flash converter inputs as shown in the Input Impedance circuit diagram. Shunting 31 parallel internal resistances of the analog switches and simultaneously charging 31 parallel 0.5 pF capacitors (only the first, 0.5 pF caps matter) is equivalent to a 400 Ohm input impedance in parallel with a 16 pF capacitor. Other input stray capacitance adds about 10 pF to the input load. Input source resistances of up to 2 kOhms can be used under normal operating conditions without any degradation of the input settling time. For larger input source resistance, longer conversion cycle times may be required to compensate for the input settling time problem. V_{REF} is set using the V_{REF+} pin.

The operation of the flash converter is divided into two parts. The first section converts the four MSBs and the second similar section converts the four LSBs. Before a conversion starts all the switches across the comparators are shut, thus forcing input and output to V_{CC}÷2. The input switches are also closed, forcing the 0.5 pF sample capacitors to track the input voltage on one side while the other side is held at V_{CC}÷2. When the switch inputs (Si) open the charge is stored on the sample capacitor. A linear resistor ladder divides the reference voltage into 32 equal steps. When the Si's open, they are connected to the stepped reference voltages at the same time the switch comparators (Sc's) are opened, allowing the input to

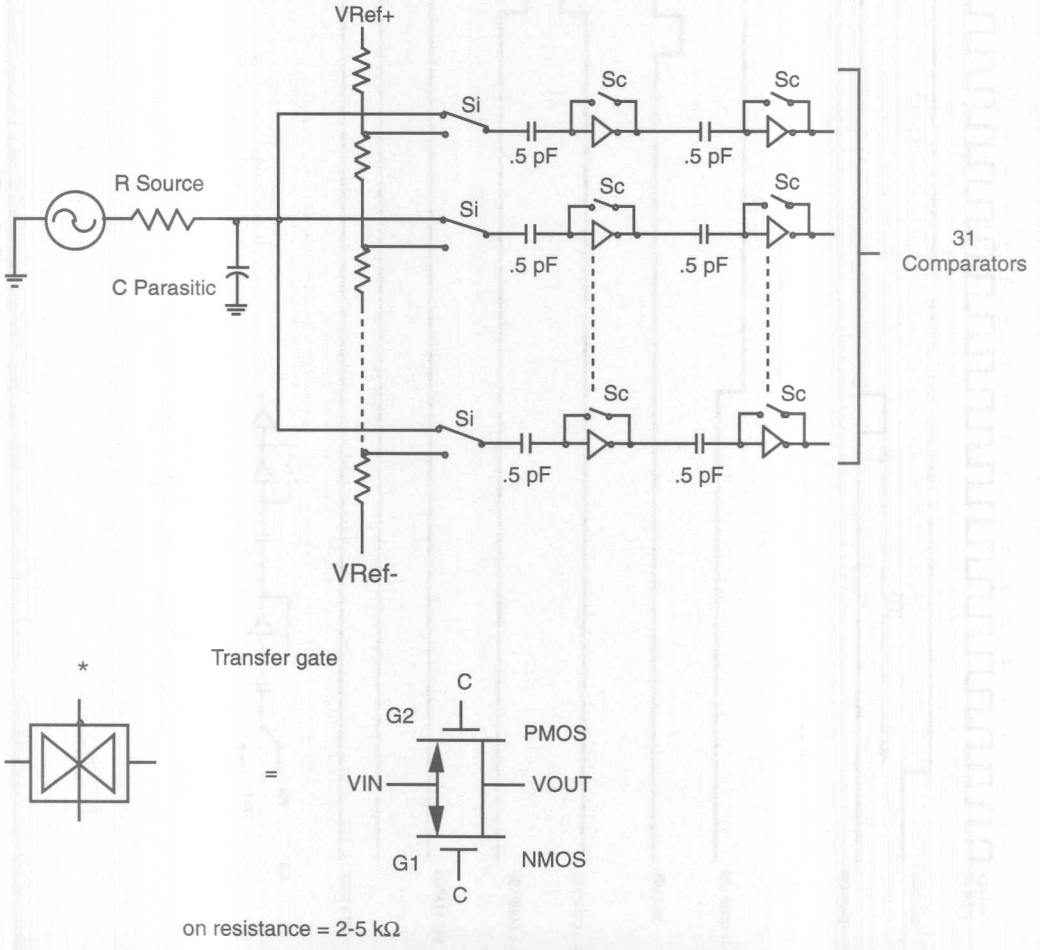


Figure 47a. Input Impedance of ADC

the comparator to change. The new input to the comparator will be the sum of the original voltage across the sample cap and the reference voltage. This voltage is compared to $V_{CC}+2$ on the threshold of the comparator. For any given input voltage, the 31 comparators will divide between all "on" above the input voltage and all "off" below it. The parallel output then is converted by logic into a binary value.

Once the determination has been made as to which point in the resistor divider the signal came closest to, the second part of the conversion takes place. In this case, the LSB part of the signal is across the resistor in the ladder adjacent to the comparison point. A second resistor ladder and comparators similar to the first are connected across the resistor. (See Figure 47b). A second conversion similar to the first takes place to complete the LSB portion.

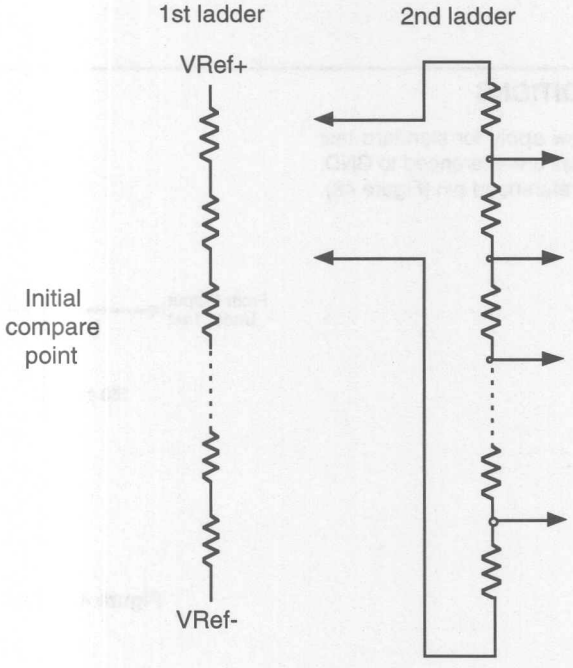


Figure 47b. Input Impedance of ADC

ABSOLUTE MAXIMUM RATINGS

Symbol	Description	Min	Max	Units
V_{CC}	Supply Voltage (*)	-0.3	+7.0	V
T_{STG}	Storage Temp	-65°	+150°	C
T_A	Oper Ambient Temp		†	C

Notes:
* Voltage on all pins with respect to GND.
† See Ordering Information.

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for an extended period may affect device reliability.

STANDARD TEST CONDITIONS

The characteristics listed below apply for standard test conditions as noted. All voltages are referenced to GND. Positive current flows into the referenced pin (Figure 48).

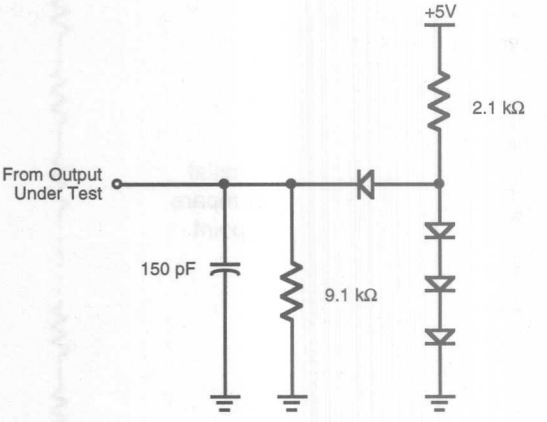


Figure 48. Test Load Diagram

CAPACITANCE

T_A = 25°C, V_{CC} = GND = 0V, f = 1.0 MHz, unmeasured pins returned to GND.

Parameter	Max
Input capacitance	12 pF
Output capacitance	12 pF
I/O capacitance	12 pF

DC ELECTRICAL CHARACTERISTICS

Sym	Parameter	V_{CC} Note [1]	T_A = 0°C to +70°C Min	Max	Typical @ 25°C	Units
I_{CC}	Supply Current	5.0V		65	40	mA
I_{CC1}	HALT Mode Current	5.0V		10	6	mA
I_{CC2}	STOP Mode Current	5.0V		20	6	µA

Notes:
[1] 5.0V ±0.5V.

DC ELECTRICAL CHARACTERISTICS

Sym	Parameter	V_{CC} Note [1]	T_A = 0°C to +70°C Min	Max	Typical @ 25°C	Units	Conditions
	Max Input Voltage	3.3V		7		V	I_{IN} 250 uA
		5.0V		7		V	I_{IN} 250 uA
V_{CH}	Clock Input High Voltage	3.3V	0.7 V_{CC}	V_{CC}+0.3	1.3	V	Driven by External Clock Generator
		5.0V	0.7 V_{CC}	V_{CC}+0.3	2.5	V	Driven by External Clock Generator
V_{CL}	Clock Input Low Voltage	3.3V	GND–0.3	0.2 V_{CC}	0.7	V	Driven by External Clock Generator
		5.0V	GND–0.3	0.2 V_{CC}	1.5	V	Driven by External Clock Generator
V_{IH}	Input High Voltage	3.3V	0.7 V_{CC}	V_{CC}+0.3	1.3	V	
		5.0V	0.7 V_{CC}	V_{CC}+0.3	2.5	V	
V_{IL}	Input Low Voltage	3.3V	GND–0.3	0.2 V_{CC}	0.7	V	
		5.0V	GND–0.3	0.2 V_{CC}	1.5	V	
V_{OH}	Output High Voltge	3.3V	V_{CC}–0.4		3.1	V	I_{OH} = –2.0 mA
		5.0V	V_{CC}–0.4		4.8	V	I_{OH} = –2.0 mA
V_{OL1}	Output Low Voltage	3.3V		0.6	0.2	V	I_{OH} = +4.0 mA
		5.0V		0.4	0.1	V	I_{OL} = +4.0 mA
V_{OL2}	Output Low Voltage	3.3V		1.2	0.3	V	I_{OL} = +6 mA, 3 Pin Max
		5.0V		1.2	0.3	V	I_{OL} = +12 mA, 3 Pin Max
V_{RH}	Reset Input High Voltage	3.3V	0.8 V_{CC}	V_{CC}	1.5	V	
		5.0V	0.8 V_{CC}	V_{CC}	2.1	V	
V_{RI}	Reset Input Low Voltage	3.3V	GND–0.3	0.2 V_{CC}	1.1		
		5.0V	GND–0.3	0.2 V_{CC}	1.7		
V_{OFFSET}	Comparator Input Offset	3.3V		25	10	mV	
	Voltage	5.0V		25	10	mV	
I_{IL}	Input Leakage	3.3V	–1	1	<1	µA	V_{IN} = 0V, V_{CC}
		5.0V	–1	1	<1	µA	V_{IN} = 0V, V_{CC}
I_{OL}	Output Leakage	3.3V	–1	1	<1	µA	V_{IN} = 0V, V_{CC}
		5.0V	–1	1	<1	µA	V_{IN} = 0V, V_{CC}
I_{IR}	Reset Input Current	3.3V		–45	–20	µA	
		5.0V		–55	–30	µA	

3

AC CHARACTERISTICS

External I/O or Memory Read and Write Timing Diagram

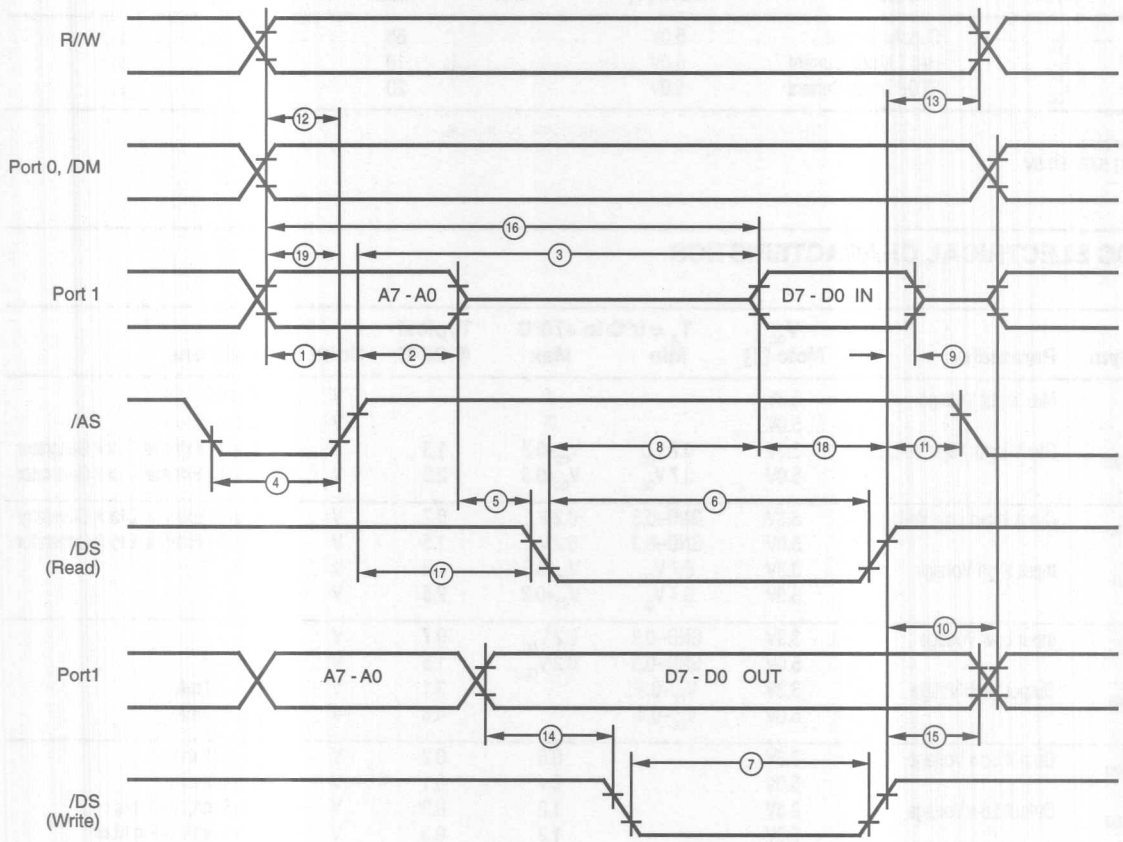


Figure 49. External I/O or Memory Read/Write Timing

AC CHARACTERISTICS

External I/O or Memory Read and Write Timing Table

No	Symbol	Parameter	V_{CC} [4]	T_A=0°C to +70°C Min	Max	Units	Notes
1	TdA(AS)	Address Valid to /AS Rise Delay	5.0V	25		ns	[2, 3]
2	TdAS(A)	/AS Rise to Address Float Delay	5.0V	35		ns	[2, 3]
3	TdAS(DR)	/AS Rise to Read Data Req'd Valid	5.0V		150	ns	[1, 2, 3]
4	TwAS	/AS Low Width	5.0V	35		ns	[2, 3]
5	TdAZ(DS)	Address Float to /DS Fall	5.0V	0		ns	
6	TwDSR	/DS (Read) Low Width	5.0V	125		ns	[1, 2, 3]
7	TwDSW	/DS (Write) Low Width	5.0V	75		ns	[1, 2, 3]
8	TdDSR(DR)	/DS Fall to Read Data Req'd Valid	5.0V		90	ns	[1, 2, 3]
9	ThDR(DS)	Read Data to /DS Rise Hold Time	5.0V	0		ns	[2, 3]
10	TdDS(A)	/DS Rise to Address Active Delay	5.0V	40		ns	[2, 3]
11	TdDS(AS)	/DS Rise to /AS Fall Delay	5.0V	35		ns	[2, 3]
12	TdR/W(AS)	R//W Valid to /AS Rise Delay	5.0V	25		ns	[2, 3]
13	TdDS(R/W)	/DS Rise to R//W Not Valid	5.0V	35		ns	[2, 3]
14	TdDW(DSW)	Write Data Valid to /DS Fall (Write) Delay	5.0V	40		ns	[2, 3]
15	TdDS(DW)	/DS Rise to Write Data Not Valid Delay	5.0V	25		ns	[2, 3]
16	TdA(DR)	Address Valid to Read Data Req'd Valid	5.0V		180	ns	[1, 2, 3]
17	TdAS(DS)	/AS Rise to /DS Fall Delay	5.0V	48		ns	[2, 3]
18	TdDI(DS)	Data Input Setup to /DS Rise	5.0V	50		ns	[1, 2, 3]
19	TdDM(AS)	/DM Valid to /AS Fall Delay	5.0V	20		ns	[2, 3]

Notes:
[1] When using extended memory timing add 2 TpC.
[2] Timing numbers given are for minimum TpC.
[3] See clock cycle dependent characteristics table.
[4] 5.0V ± 0.5V.

Standard Test Load
All timing references use 0.9 V_{CC} for a logic1 and 0.1 V_{CC} for a logic0.

AC ELECTRICAL CHARACTERISTICS
Additional Timing Diagram

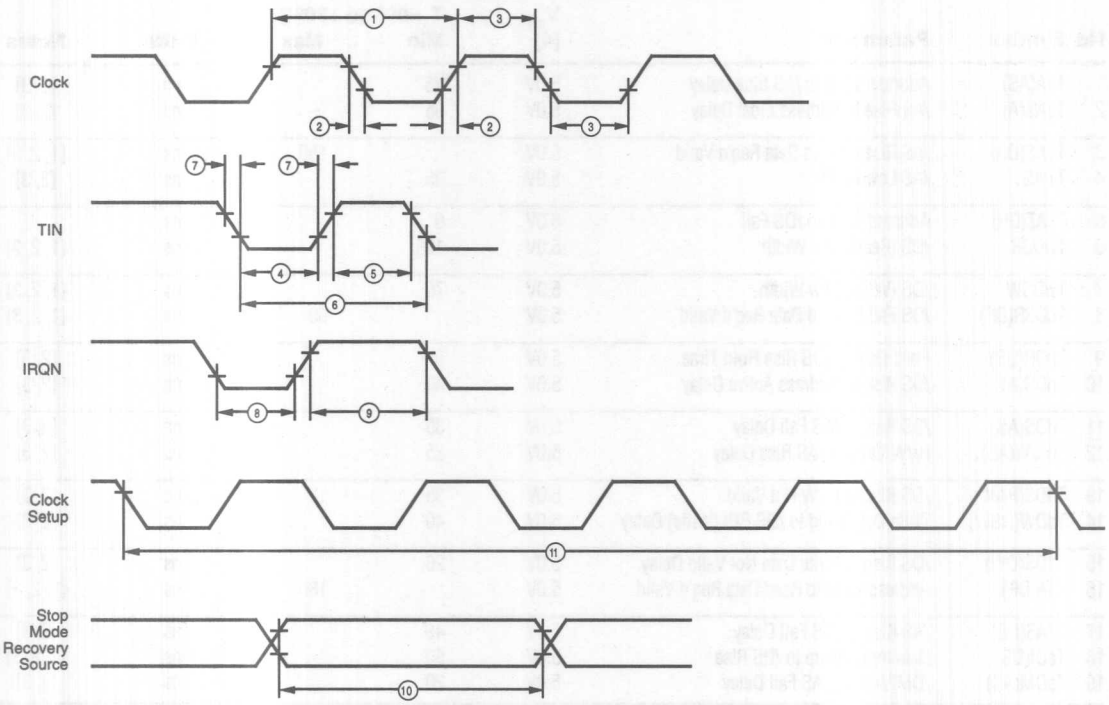


Figure 50. Additional Timing

AC ELECTRICAL CHARACTERISTICS
Additional Timing Table

No	Symbol	Parameter	V_{CC} Note [5]	T_A=0°C to +70°C Min	Max	Units	Notes
1	TpC	Input Clock Period	5.0V	48.83		ns	[1]
2	TrC,TfC	Clock Input Rise & Fall Times	5.0V		6	ns	[1]
3	TwC	Input Clock Width	5.0V	16		ns	[1]
4	TwTinL	Timer Input Low Width	5.0V	70		ns	
5	TwTinH	Timer Input High Width	5.0V	3TpC			[1]
6	TpTin	Timer Input Period	5.0V	8TpC			[1]
7	TrTin, TfTin	Timer Input Rise & Fall Timer	5.0V		100	ns	[1]
8A	TwIL	Int. Request Low Time	5.0V	70		ns	[1, 2]
8B	TwIL	Int. Request Low Time	5.0V	3TpC			[1]
9	TwIH	Int. Request Input High Time	5.0V	3TpC			[1]
10	Twsm	Stop-Mode Recovery Width Spec	5.0V	12 5TpC		ns	[1]
11	Tost	Oscillator Startup Time	5.0V	5TpC			[3]
12	Twdt	Watch-Dog Timer	5.0V	5		ms	D0 = 0 [4]
			5.0V	15		ms	D0 = 1 [4]
			5.0V	25		ms	D0 = 0 [4]
			5.0V	100		ms	D0 = 1 [4]

Notes:
[1] Timing Reference uses 0.9 V_{CC} for a logic1 and 0.1 V_{CC} for a logic0.
[2] Interrupt request via Port 3 (P31-P33).
[3] SMR-D5 = 0.
[4] Reg. WDT.
[5] 5.0V ± 0.5V.

3

AC ELECTRICAL CHARACTERISTICS
Handshake Timing Diagrams

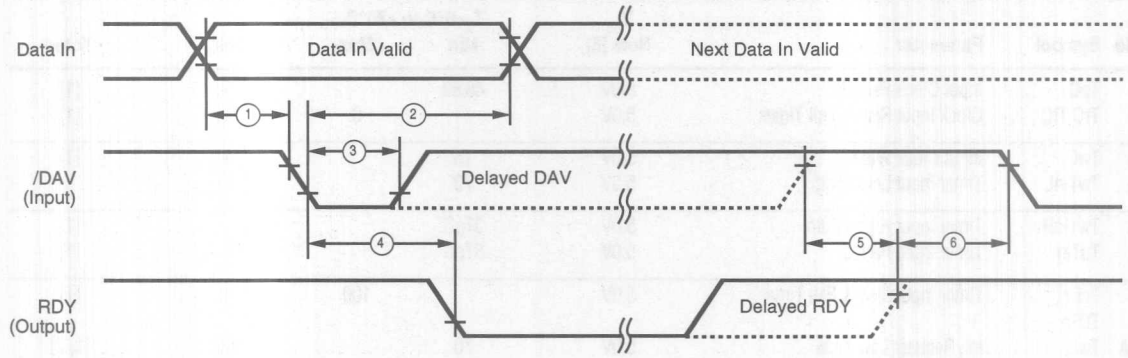


Figure 51. Input Handshake Timing

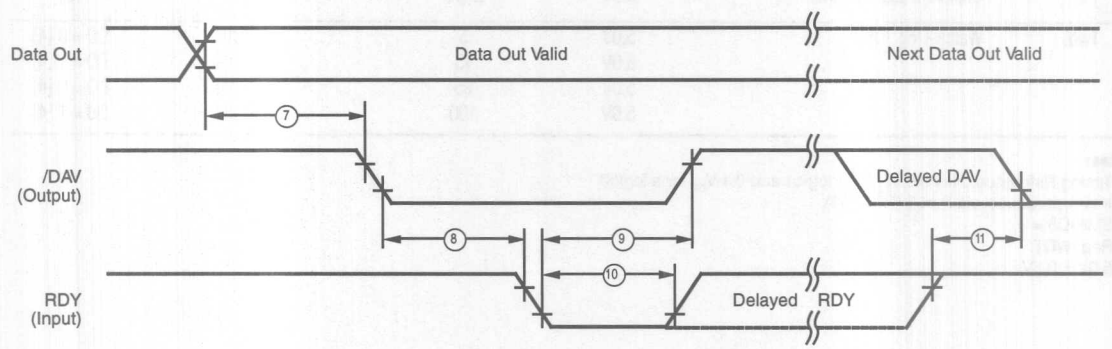


Figure 52. Output Handshake Timing

AC ELECTRICAL CHARACTERISTICS
Handshake Timing Table

No	Symbol	Parameter	V_{CC} Note [1]	T_A=0°C to +70°C Min	Max	Units	Data Direction
1	TsDI(DAV)	Data In Setup Time	5.0V	0		ns	IN
2	ThDI(DAV)	Data In Hold Time	5.0V	115		ns	IN
3	TwDAV	Data Available Width	5.0V	110		ns	IN
4	TdDAVI(RDY)	DAV Fall to RDY Fall Delay	5.0V		115	ns	IN
5	TdDAVId(RDY)	DAV Rise to RDY Rise Delay	5.0V		80	ns	IN
6	TdDO(DAV)	RDY Rise to DAV Fall Delay	5.0V	0		ns	IN
7	TcLDAV0(RDY)	Data Out to DAV Fall Delay	5.0V	25		ns	OUT
8	TcLDAV0(RDY)	DAV Fall to RDY Fall Delay	5.0V	0		ns	OUT
9	TdRDY0(DAV)	RDY Fall to DAV Rise Delay	5.0V		115	ns	OUT
10	TwRDY	RDY Width	5.0V	80		ns	OUT
11	TdRDY0d(DAV)	RDY Rise to DAV Fall Delay	5.0V		80	ns	OUT

Note:
[1] 5.0V ± 0.5V

3

ELECTRICAL CHARACTERISTICS

A/D Electrical Characteristics

T_A = 0°C - 70°C; V_{cc} = 5.0V ± 0.5V

Parameter	Minimum	Maximum	Typical	Units
Resolution			8	bits
Integral non-linearity		1	0.5	lsb
Differential non-linearity		0.5		lsb
Zero Error at 25°C		50		mV
Power Dissipation		75	35	mW
Clock Frequency		20		MHz
Clock Pulse Width	35			ns
Input Voltage Range	AN_{GND}	ANV_{CC}		V
Conversion Time		2		µs
Input Capacitance on		60		pF
VA_{HI} range damage	AN_{GND}	ANV_{CC}		V
VA_{LO} range damage	AN_{GND}	ANV_{CC}		V
AN_{GND}	V_{SS}	ANV_{CC}		V
ANV_{CC}	AN_{GND}	V_{CC}		V
III ana	–10	+10		µA
III VA_{HI}, VA_{LO}	TBD	TBD		µA

Z8 EXPANDED REGISTER FILE REGISTERS

Expanded Register Bank B

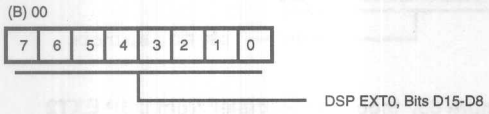


Figure 53. Outgoing Register to DSP EXT0 (High Byte) (B) 00H [Read/Write]

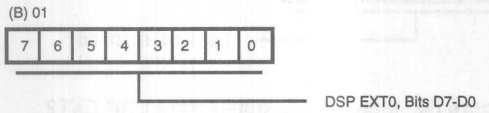


Figure 54. Outgoing Register to DSP EXT0 (Low Byte) (B) 01H [Read/Write]

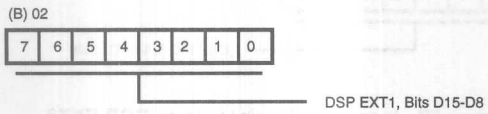


Figure 55. Outgoing Register to DSP EXT1 (High Byte) (B) 02H [Read/Write]

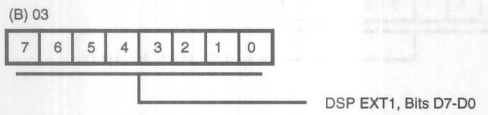


Figure 56. Outgoing Register to DSP EXT1 (Low Byte) (B) 03H [Read/Write]

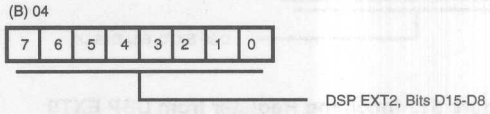


Figure 57. Outgoing Register to DSP EXT2 (High Byte) (B) 04H [Read/Write]

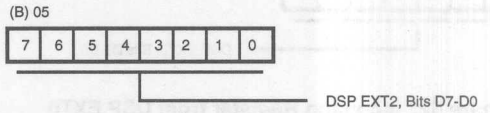


Figure 58. Outgoing Register to DSP EXT2 (Low Byte) (B) 05H [Read/Write]

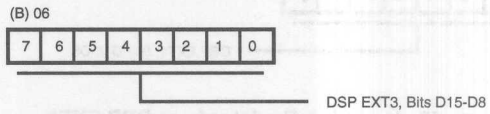


Figure 59. Outgoing Register to DSP EXT3 (High Byte) (B) 06H [Read/Write]

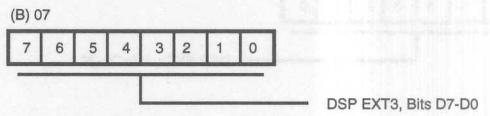


Figure 60. Outgoing Register to DSP EXT3 (Low Byte) (B) 07H [Read/Write]

3

Expanded Register Bank B (Continued)

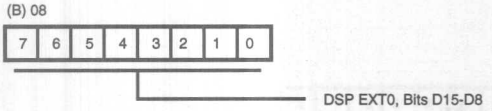


Figure 61. Incoming Register from DSP EXT0 (High Byte) (B) 08H [Read Only]

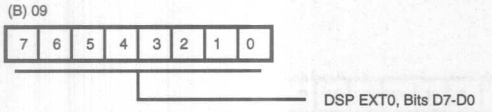


Figure 62. Incoming Register from DSP EXT0 (Low Byte) (B) 09H [Read Only]

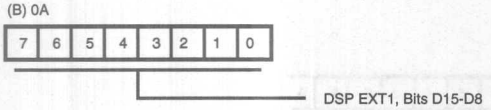


Figure 63. Incoming Register from DSP EXT1 (High Byte) (B) 0AH [Read Only]

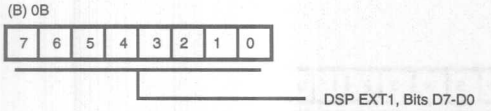


Figure 64. Incoming Register from DSP EXT1 (Low Byte) (B) 0BH [Read Only]

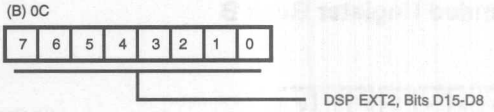


Figure 65. Incoming Register from DSP EXT2 (High Byte) (B) 0CH [Read Only]

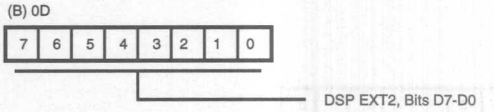


Figure 66. Incoming Register from DSP EXT2 (Low Byte) (B) 0DH [Read Only]

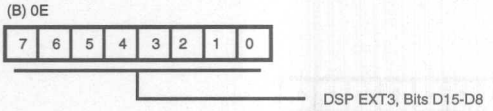


Figure 67. Incoming Register from DSP EXT3 (High Byte) (B) 0EH [Read Only]

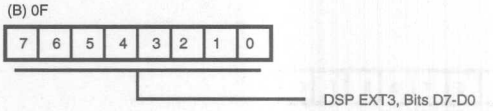


Figure 68. Incoming Register from DSP EXT3 (Low Byte) (B) 0FH [Read Only]

Expanded Register Bank F

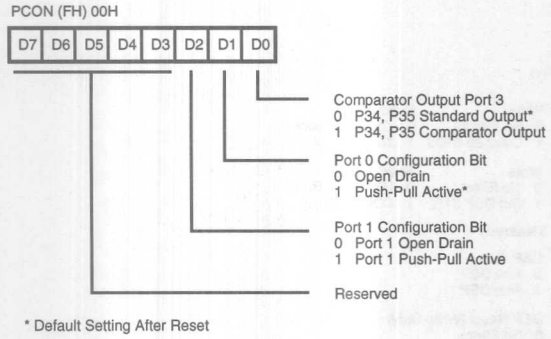


Figure 69. Port Configuration Register (PCON) (F) 00 [Write Only]

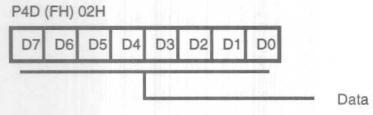


Figure 70. Port 4 Data Register (F) 02H [Read/Write]

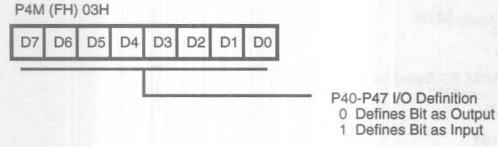


Figure 71. Port 4 Mode Register (F) 03H [Write Only]

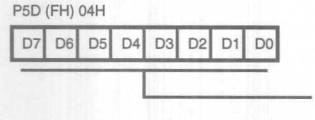


Figure 72. Port 5 Data Register (F) 04H [Read/Write]

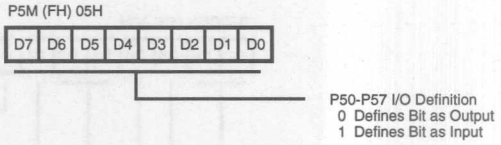


Figure 73. Port 5 Mode Register (F) 05H [Write Only]

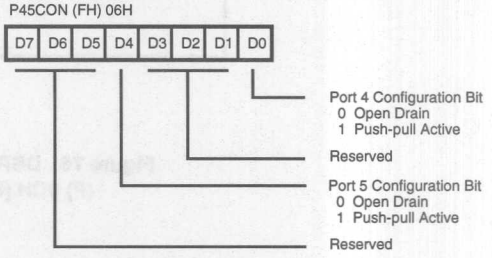


Figure 74. Port 4 and 5 Configuration Register (F) 06H [Write Only]

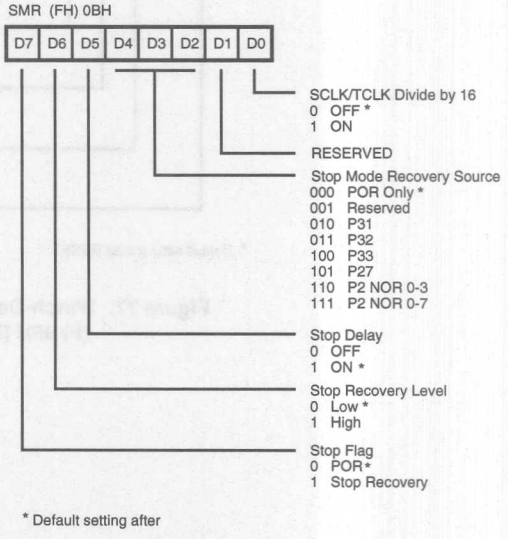


Figure 75. Stop-Mode Recovery Register (SMR) (F) 0BH [Read/Write]

3

Expanded Register Bank F (Continued)

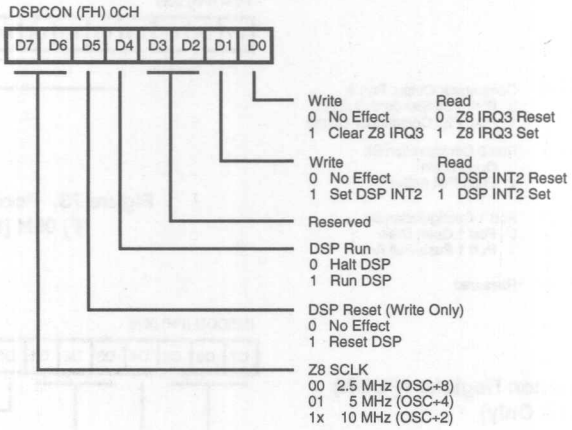


Figure 76. DSP Control Register (F) 0CH [Read/Write]

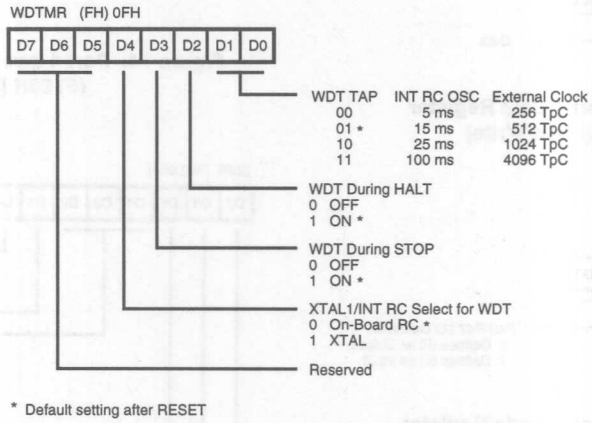


Figure 77. Watch-Dog Timer Mode Register (F) 0FH [Read/Write]

Z8 CONTROL REGISTERS

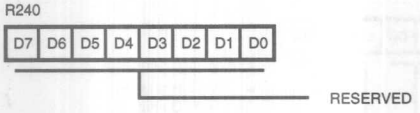


Figure 78. Reserved (F0H)

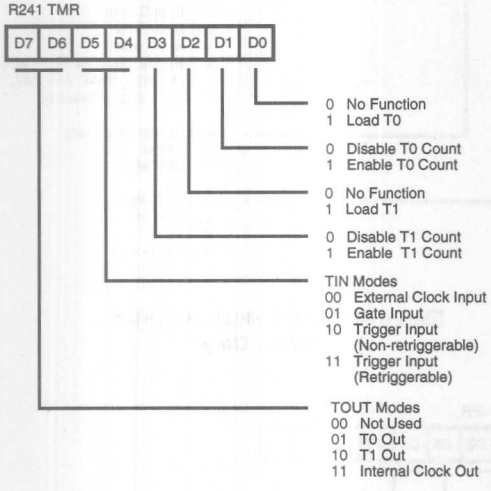


Figure 79. Timer Mode Register (F1H:Read/Write)

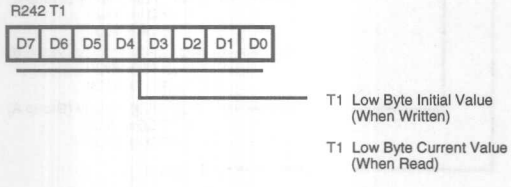


Figure 80. Counter/Timer 1 Register (F2H:Read/Write)

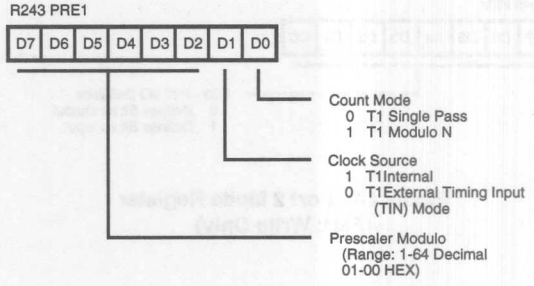


Figure 81. Prescaler 1 Register (F3H:Write Only)

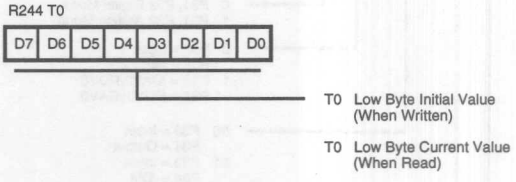


Figure 82. Counter/Timer 0 Register (F4H:Read/Write)

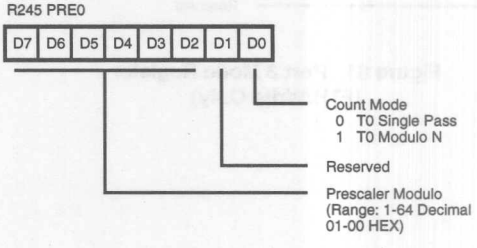


Figure 83. Prescaler 0 Register (F5H:Write Only)

3

Z8 CONTROL REGISTERS (Continued)

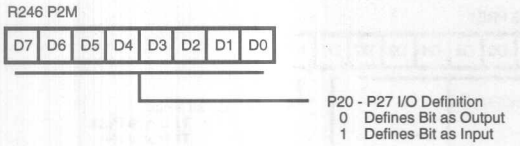


Figure 84. Port 2 Mode Register (F6H: Write Only)

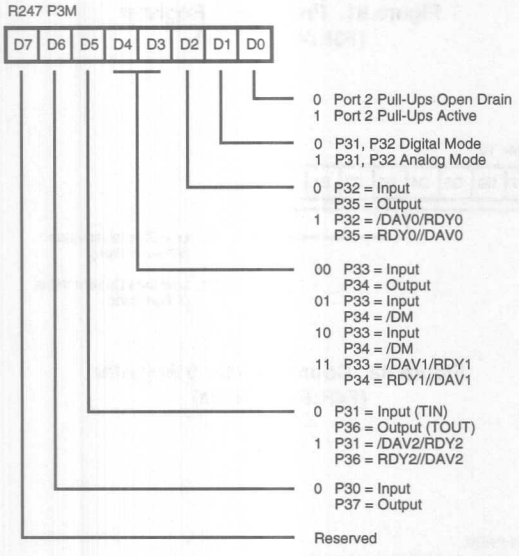


Figure 85. Port 3 Mode Register (F7H:Write Only)

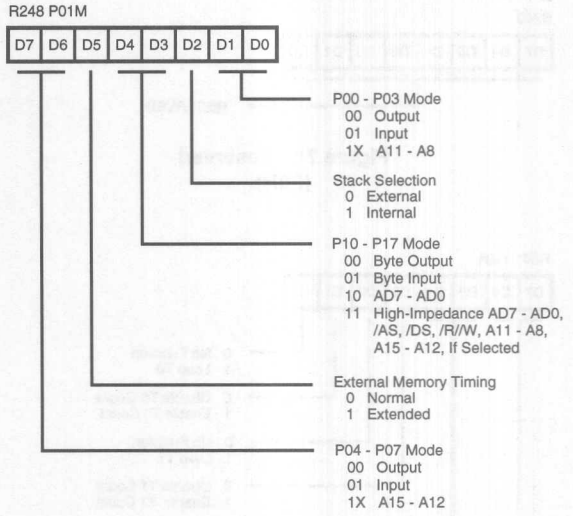


Figure 86. Port 0 Mode Register (F8H:Write Only)

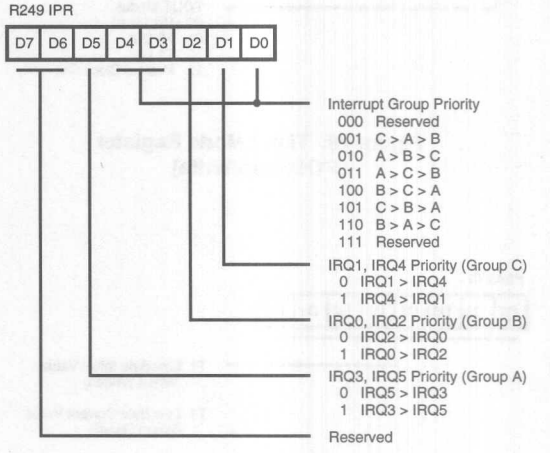


Figure 87. Interrupt Priority Register (F9H:Write Only)

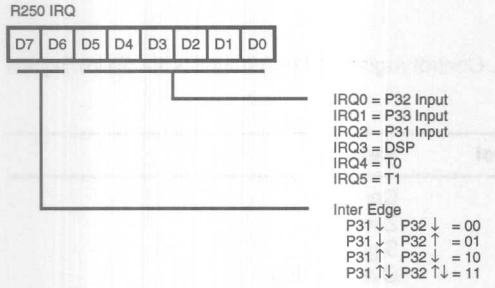


Figure 88. Interrupt Request Register (FAH:Read/Write)

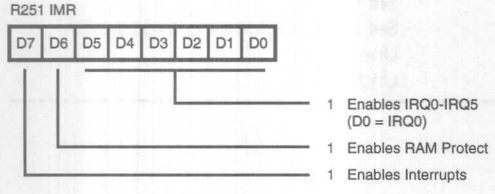


Figure 89. Interrupt Mask Register (FBH:Read/Write)

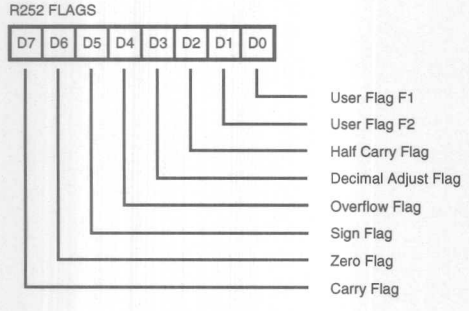


Figure 90. Flag Register (FCH:Read/Write)

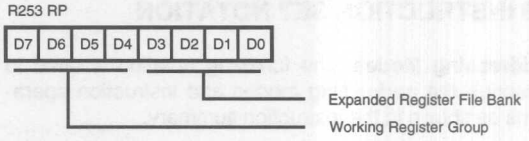


Figure 91. Register Pointer (FDH:Read/Write)

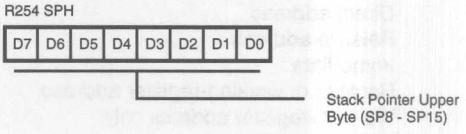


Figure 92. Stack Pointer High (FEH:Read/Write)

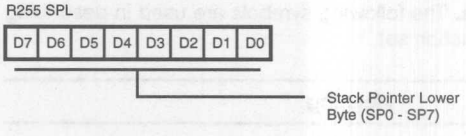


Figure 93. Stack Pointer Low (FFH:Read/Write)

3

Z8 INSTRUCTION SET NOTATION

Addressing Modes. The following notation is used to describe the addressing modes and instruction operations as shown in the instruction summary.

Symbol	Meaning
IRR	Indirect register pair or indirect working-register pair address
Irr	Indirect working-register pair only
X	Indexed address
DA	Direct address
RA	Relative address
IM	Immediate
R	Register or working-register address
r	Working-register address only
IR	Indirect-register or indirect working-register address
Ir	Indirect working-register address only
RR	Register pair or working register pair address

Symbols. The following symbols are used in describing the instruction set.

Symbol	Meaning
dst	Destination location or contents
src	Source location or contents
cc	Condition code
@	Indirect address prefix
SP	Stack Pointer
PC	Program Counter
FLAGS	Flag register (Control Register 252)
RP	Register Pointer (R253)
IMR	Interrupt mask register (R251)

Flags. Control register (R252) contains the following six flags:

Symbol	Meaning
C	Carry flag
Z	Zero flag
S	Sign flag
V	Overflow flag
D	Decimal-adjust flag
H	Half-carry flag

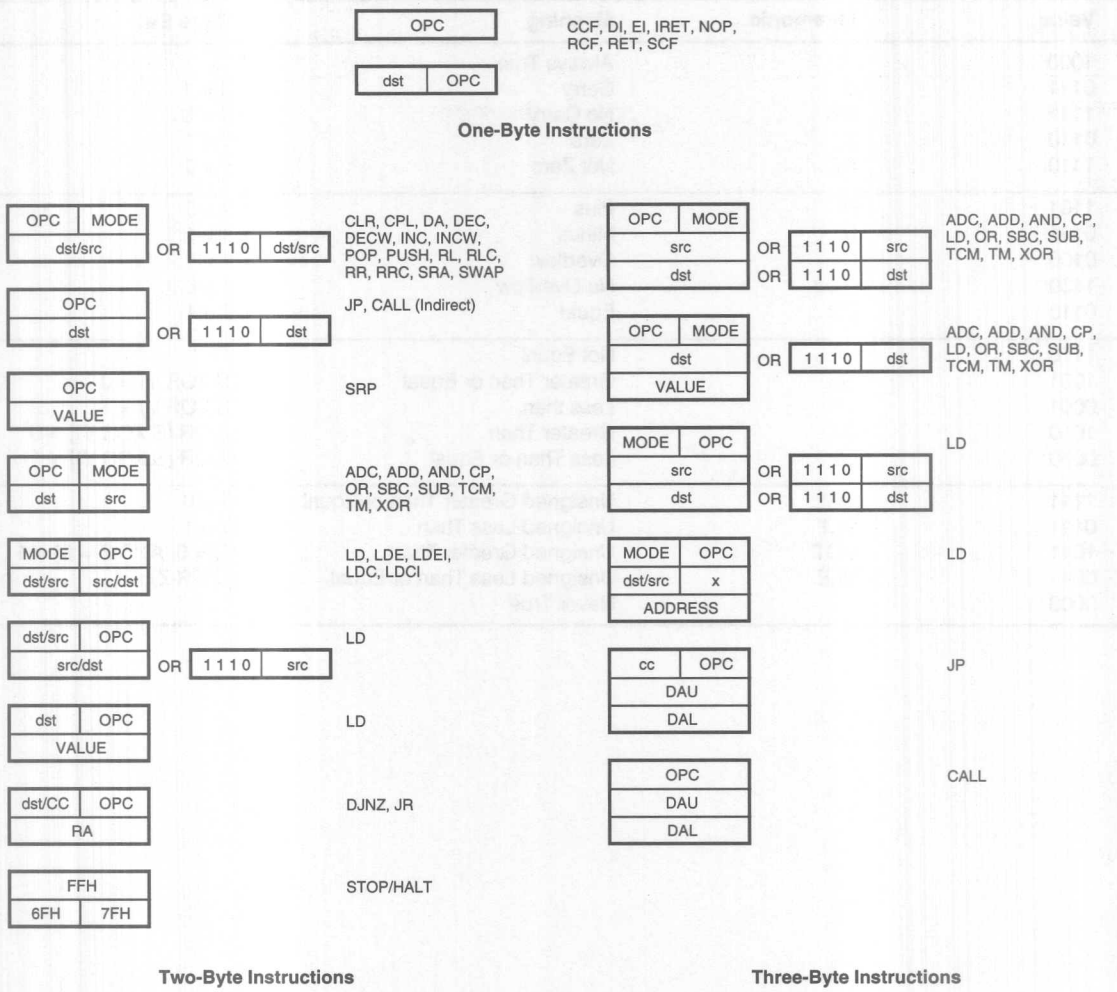
Affected flags are indicated by:

0	Clear to zero
1	Set to one
*	Set to clear according to operation
-	Unaffected
x	Undefined

CONDITION CODES

Value	Mnemonic	Meaning	Flags Set
1000		Always True	
0111	C	Carry	C = 1
1111	NC	No Carry	C = 0
0110	Z	Zero	Z = 1
1110	NZ	Not Zero	Z = 0
1101	PL	Plus	S = 0
0101	MI	Minus	S = 1
0100	OV	Overflow	V = 1
1100	NOV	No Overflow	V = 0
0110	EQ	Equal	Z = 1
1110	NE	Not Equal	Z = 0
1001	GE	Greater Than or Equal	(S XOR V) = 0
0001	LT	Less than	(S XOR V) = 1
1010	GT	Greater Than	[Z OR (S XOR V)] = 0
0010	LE	Less Than or Equal	[Z OR (S XOR V)] = 1
1111	UGE	Unsigned Greater Than or Equal	C = 0
0111	ULT	Unsigned Less Than	C = 1
1011	UGT	Unsigned Greater Than	(C = 0 AND Z = 0) = 1
0011	ULE	Unsigned Less Than or Equal	(C OR Z) = 1
0000		Never True	

INSTRUCTION FORMATS



INSTRUCTION SUMMARY

Note: Assignment of a value is indicated by the symbol " ← ". For example:

dst ← dst + src

indicates that the source data is added to the destination data and the result is stored in the destination location. The notation "addr (n)" is used to refer to bit (n) of a given operand location. For example:

dst (7)

refers to bit 7 of the destination operand.

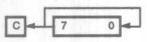
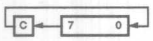
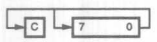
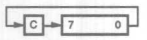
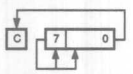
INSTRUCTION SUMMARY (Continued)

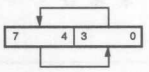
Instruction and Operation	Address Mode dst	Address Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
ADC dst, src dst←dst + src +C	†		1[ ]	*	*	*	*	0	*
ADD dst, src dst←dst + src	†		0[ ]	*	*	*	*	0	*
AND dst, src dst←dst AND src	†		5[ ]	-	*	*	0	-	-
CALL dst SP←SP - 2 @SP←PC, PC←dst	DA IRR		D6 D4	-	-	-	-	-	-
CCF C←NOT C			EF	*	-	-	-	-	-
CLR dst dst←0	R IR		B0 B1	-	-	-	-	-	-
COM dst dst←NOT dst	R IR		60 61	-	*	*	0	-	-
CP dst, src dst - src	†		A[ ]	*	*	*	*	-	-
DA dst dst←DA dst	R IR		40 41	*	*	*	X	-	-
DEC dst dst←dst - 1	R IR		00 01	-	*	*	*	-	-
DECW dst dst←dst - 1	RR IR		80 81	-	*	*	*	-	-
DI IMR(7)←0			8F	-	-	-	-	-	-
DJNZ dst r←r - 1 if r ≠ 0 PC←PC + dst Range: +127, -128	RA		rA r = 0 - F	-	-	-	-	-	-
EI IMR(7)←1			9F	-	-	-	-	-	-
HALT			7F	-	-	-	-	-	-

Instruction and Operation	Address Mode dst	Address Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
INC dst dst←dst + 1	r		rE r = 0 - F	-	*	*	*	-	-
	R		20						
	IR		21						
INCW dst dst←dst + 1	RR IR		A0 A1	-	*	*	*	-	-
IRET FLAGS←@SP; SP←SP + 1 PC←@SP; SP←SP + 2; IMR(7)←1			BF	*	*	*	*	*	*
JP cc, dst if cc is true PC←dst	DA IRR		cD c = 0 - F 30	-	-	-	-	-	-
JR cc, dst if cc is true, PC←PC + dst Range: +127, -128	RA		cB c = 0 - F	-	-	-	-	-	-
LD dst, src dst←src	r	Im	rC	-	-	-	-	-	-
	r	R	r8						
	R	r	r9						
			r = 0 - F						
	r	X	C7						
	X	r	D7						
	r	Ir	E3						
	Ir	r	F3						
	R	R	E4						
	R	IR	E5						
	R	IM	E6						
	IR	IM	E7						
	IR	R	F5						
LDC dst, src	r	Irr	C2	-	-	-	-	-	-
LDCI dst, src dst←src r←r +1; rr←rr + 1	Ir	Irr	C3	-	-	-	-	-	-

3

INSTRUCTION SUMMARY (Continued)

Instruction and Operation	Address Mode dst	Address Mode src	Opcode Byte (Hex)	C	Z	S	V	D	H
NOP			FF	-	-	-	-	-	-
OR dst, src dst←dst OR src	†		4[ ]	-	*	*	0	-	-
POP dst dst←@SP; SP←SP + 1	R IR		50 51	-	-	-	-	-	-
PUSH src SP←SP - 1; @SP←src		R IR	70 71	-	-	-	-	-	-
RCF C←0			CF	0	-	-	-	-	-
RET PC←@SP; SP←SP + 2			AF	-	-	-	-	-	-
RL dst 	R IR		90 91	*	*	*	*	-	-
RLC dst 	R IR		10 11	*	*	*	*	-	-
RR dst 	R IR		E0 E1	*	*	*	*	-	-
RRC dst 	R IR		C0 C1	*	*	*	*	-	-
SBC dst, src dst←dst←src←C	†		3[ ]	*	*	*	*	1	*
SCF C←1			DF	1	-	-	-	-	-
SRA dst 	R IR		D0 D1	*	*	*	0	-	-
SRP src RP←src		Im	31	-	-	-	-	-	-

Instruction and Operation	Address Mode dst	Address Mode src	Opcode Byte (Hex)	C	Z	S	V	D	H
STOP			6F	-	-	-	-	-	-
SUB dst, src dst←dst←src	†		2[ ]	*	*	*	*	1	*
SWAP dst 	R IR		F0 F1	X	*	*	X	-	-
TCM dst, src (NOT dst) AND src	†		6[ ]	-	*	*	0	-	-
TM dst, src dst AND src	†		7[ ]	-	*	*	0	-	-
XOR dst, src dst←dst XOR src	†		B[ ]	-	*	*	0	-	-

† These instructions have an identical set of addressing modes, which are encoded for brevity. The first opcode nibble is found in the instruction set table above. The second nibble is expressed symbolically by a '[ ]' in this table, and its value is found in the following table to the left of the applicable addressing mode pair.

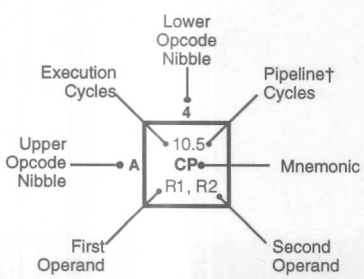
For example, the opcode of an ADC instruction using the addressing modes r (destination) and Ir (source) is 13.

Address Mode dst	Address Mode src	Lower Opcode Nibble
r	r	[2]
r	Ir	[3]
R	R	[4]
R	IR	[5]
R	IM	[6]
IR	IM	[7]

OPCODE MAP

Lower Nibble (Hex)

Upper Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	6.5 **DEC** R1	6.5 **DEC** IR1	6.5 **ADD** r1, r2	6.5 **ADD** r1, Ir2	10.5 **ADD** R2, R1	10.5 **ADD** IR2, R1	10.5 **ADD** R1, IM	10.5 **ADD** IR1, IM	6.5 **LD** r1, R2	6.5 **LD** r2, R1	12/10.5 **DJNZ** r1, RA	12/10.0 **JR** cc, RA	6.5 **LD** r1, IM	12.10.0 **JP** cc, DA	6.5 **INC** r1	
1	6.5 **RLC** R1	6.5 **RLC** IR1	6.5 **ADC** r1, r2	6.5 **ADC** r1, Ir2	10.5 **ADC** R2, R1	10.5 **ADC** IR2, R1	10.5 **ADC** R1, IM	10.5 **ADC** IR1, IM								
2	6.5 **INC** R1	6.5 **INC** IR1	6.5 **SUB** r1, r2	6.5 **SUB** r1, Ir2	10.5 **SUB** R2, R1	10.5 **SUB** IR2, R1	10.5 **SUB** R1, IM	10.5 **SUB** IR1, IM								
3	8.0 **JP** IRR1	6.1 **SRP** IM	6.5 **SBC** r1, r2	6.5 **SBC** r1, Ir2	10.5 **SBC** R2, R1	10.5 **SBC** IR2, R1	10.5 **SBC** R1, IM	10.5 **SBC** IR1, IM								
4	8.5 **DA** R1	8.5 **DA** IR1	6.5 **OR** r1, r2	6.5 **OR** r1, Ir2	10.5 **OR** R2, R1	10.5 **OR** IR2, R1	10.5 **OR** R1, IM	10.5 **OR** IR1, IM								
5	10.5 **POP** R1	10.5 **POP** IR1	6.5 **AND** r1, r2	6.5 **AND** r1, Ir2	10.5 **AND** R2, R1	10.5 **AND** IR2, R1	10.5 **AND** R1, IM	10.5 **AND** IR1, IM								
6	6.5 **COM** R1	6.5 **COM** IR1	6.5 **TCM** r1, r2	6.5 **TCM** r1, Ir2	10.5 **TCM** R2, R1	10.5 **TCM** IR2, R1	10.5 **TCM** R1, IM	10.5 **TCM** IR1, IM								6.0 **STOP**
7	10/12.1 **PUSH** R2	12/14.1 **PUSH** IR2	6.5 **TM** r1, r2	6.5 **TM** r1, Ir2	10.5 **TM** R2, R1	10.5 **TM** IR2, R1	10.5 **TM** R1, IM	10.5 **TM** IR1, IM								7.0 **HALT**
8	10.5 **DECW** RR1	10.5 **DECW** IR1	12.0 **LDE** r1, Irr2	18.0 **LDEI** Ir1, Irr2												6.1 **DI**
9	6.5 **RL** R1	6.5 **RL** IR1	12.0 **LDE** r2, Irr1	18.0 **LDEI** Ir2, Irr1												6.1 **EI**
A	10.5 **INCW** RR1	10.5 **INCW** IR1	6.5 **CP** r1, r2	6.5 **CP** r1, Ir2	10.5 **CP** R2, R1	10.5 **CP** IR2, R1	10.5 **CP** R1, IM	10.5 **CP** IR1, IM								14.0 **RET**
B	6.5 **CLR** R1	6.5 **CLR** IR1	6.5 **XOR** r1, r2	6.5 **XOR** r1, Ir2	10.5 **XOR** R2, R1	10.5 **XOR** IR2, R1	10.5 **XOR** R1, IM	10.5 **XOR** IR1, IM								16.0 **IRET**
C	6.5 **RRC** R1	6.5 **RRC** IR1	12.0 **LDC** r1, Irr2	18.0 **LDCI** Ir1, Irr2				10.5 **LD** r1,x,R2								6.5 **RCF**
D	6.5 **SRA** R1	6.5 **SRA** IR1	12.0 **LDC** r2, Irr1	18.0 **LDCI** Ir2, Irr1	20.0 **CALL*** IRR1		20.0 **CALL** DA	10.5 **LD** r2,x,R1								6.5 **SCF**
E	6.5 **RR** R1	6.5 **RR** IR1		6.5 **LD** r1, IR2	10.5 **LD** R2, R1	10.5 **LD** IR2, R1	10.5 **LD** R1, IM	10.5 **LD** IR1, IM								6.5 **CCF**
F	8.5 **SWAP** R1	8.5 **SWAP** IR1		6.5 **LD** Ir1, r2		10.5 **LD** R2, IR1										6.0 **NOP**
Bytes per Instruction	2				3				2			3			1	



Legend:
R = 8-bit address
r = 4-bit address
R^1 or r^2 = Dst address
R^1 or r^2 = Src address

Sequence:
Opcode, First Operand, Second Operand

Note: The blank areas are not defined.

* 2-byte instruction appears as a 3-byte instruction

† Refer to page 16 for pipeline instructions.

3

DSP INSTRUCTION SET NOTATION

Register Names. The following lists the register names and their descriptions.

Name	Description
A	Accumulator
BUS	Bus Dummy Register
Dn:b	Data Register where n is the register number (0...3) and b is the bank in which is resides (0..1).
EXTn	Extended Registers where n is the register number (0..7).
P	Multiplier Product Register
Pn:b	Pointer Registers where n is the register number (0...2) and b is the bank into which it points (0...1).
X	Multiplier Input Register X
Y	Multiplier Input Register Y
PC	Program Counter Register
SR	Status Register

Condition Codes. The following defines the condition codes supported by the DSP assembler. In the instruction descriptions, condition codes are referred to via the <cc> symbol. If the instruction description refers to a condition code in one of its addressing modes, the instruction will only execute if the condition is true.

Name	Description
C	Carry
EQ	Equal (same as Z)
F	False
IE	Interrupts Enabled
MI	Minus
NC	No Carry
NE	Not Equal (same as NZ)
NIE	Not Interrupts Enabled
NOV	Not Overflow
NU0	Not User Zero
NU1	Not User One
NZ	Not zero
OV	Overflow
PL	Plus (Positive)
U0	User Zero
U1	User One
UGE	Unsigned Greater Than or Equal (Same as NC)
ULT	Unsigned Less Than (Same as C)
Z	Zero

Bank Switch Enumerations. The third (optional) operand of the MLD, MPYA and MPYS instructions represents whether a bank switch is set on or off. To more clearly represent this two keywords are used (ON and OFF) which state the direction of the switch. These keywords are refered to in the instruction descriptions via the <bank switch> symbol.

Addressing Modes. This section discusses the syntax of the addressing modes supported by the DSP assembler. The symbolic name is used in the discussion of instruction syntax in the instruction descriptions.

Symbolic Name	Syntax	Description
<pregs>	Pn:b	Pointer Register
<dregs> (Points toRAM)	Dn:b	Data Register
<hwregs>	X,Y,PC,SR,P EXTn,A,BUS	Hardware Registers
<accind> (Points to Program Memory)	@A	Accumulator Memory Indirect
<direct>	<expression>	Direct Address Expression
<limm>	#<const exp>	Word (16-bit) Immediate Value
<simm>	#<const exp>	Short (8-bit) Immediate Value
<regind> (Points to RAM)	@Pn:b @Pn:b-LOOP @Pn:b+LOOP	Pointer Register Indirect Pointer Register Indirect with Loop Decrement Pointer register Indirect with Loop Increment
<memind> (Points to Program Memory	@@Pn:d @Dn:b @@Pn:b-LOOP @@Pn:b+LOOP @@Pn:b+	Pointer Register Memory Indirect Data Register Memory Indirect Pointer Register Memory Indirect with Loop Decrement Pointer Register Memory Indirect with Loop Increment Pointer Register Memory Indirect with Increment

DSP INSTRUCTION DESCRIPTIONS

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
ABS	Absolute Value	ABS[<cc>,]<src>	<cc>,A	1	1	ABS NC,A
			A	1	1	ABS A
ADD	Addition	ADD<dest>,<src>	A,<pregs>	1	1	ADD A,#128
			A,<dregs>	1	1	ADD A,D0:1
			A,<limm>	2	2	ADD A,@@LOOP
			A,<memind>	1	3	ADD A,@P2:1+
			A,<direct>	1	1	ADD A,X
			A,<regind>	1	1	
			A,<hwregs>	1	1	
AND	Bitwise AND	AND<dest>,<src>	A,<pregs>	1	1	AND A,#128
			A,<dregs>	1	1	AND A,D0:1
			A,<limm>	2	2	AND A,@@P0:0+LOOP
			A,<memind>	1	3	AND A,@P2:1+
			A,<direct>	1	1	
			A,<regind>	1	1	AND A,X
			A,<hwregs>	1	1	
CALL	Subroutine call	CALL [<cc>,]<address>	<cc>,<direct>	2	2	CALL sub1
			<direct>	2	2	CALL Z,sub2
CCF	Clear carry flag	CCF	None	1	1	CCF
CIEF	Clear Carry Flag	CIEF	None	1	1	CIEF
COPF	Clear OP flag	COPF	None	1	1	COPF
CP	Comparison	CP<src1>,<src2>	A,<pregs>	1	1	CP A,P0:0
			A,<dregs>	1	1	CP A,D3:1
			A,<memind>	1	3	CP A,#512
			A,<direct>	1	1	CP A,@@P0:1
			A,<regind>	1	1	CP A,LABEL
			A,<hwregs>	1	1	CP A,@D0:0
						CP A,X
DEC	Decrement	DEC [<cc>,]<dest>	<cc>A,	1	1	DEC NZ,A
			A	1	1	DEC A
INC	Increment	INC [<cc>,] <dest>	<cc>,A	1	1	INC NZ,A
			A	1	1	INC A
JP	Jump	JP [<cc>,]<address>	<cc>,<direct>	2	2	JP NIE,Label
			<direct>	2	2	JP Label

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
LD	Load destination with source	LD<dest>,<src>	A,<hwregs>	1	1	LD A,X
			A,<dregs>	1	1	LD A,D0:0
			A,<pregs>	1	1	LD A,P0:1
			A,<regind>	1	1	LD A,@@P1:1
			A,<memind>	1	3	LD A,MEMADDR
			A,<direct>	1	1	LD MEMADDR,A
			<direct>,A	1	1	LD D0:1,A
			<dregs>,<hwregs>	1	1	LD P1:0#128
			<pregs>,<simm>	1	1	LDP1:1,X
			<pregs>,<hwregs>	1	1	LD@P0:0+LOOP,#1234
			<regind>,<limm>	1	1	LD @P1:1+,X
			<regind>,<hwregs>	1	1	LD X,P0:0
			<hwregs>,<pregs>	1	1	LD Y,P0:0
			<hwregs>,<dregs>	1	1	LD SR,#%1023
			<hwregs>,<limm>	2	2	LD PC,(A)
			<hwregs>,<accind>	1	3	LD X,@@P0:0
			<hwregs>,<memind>	1	3	LD Y,@P1:0–LOOP
			<hwregs>,<regind>	1	1	LD SR,X
			<hwregs>,<hwregs>	1	1	
			Note: When <dest> is <hwregs>, <dest> cannot be P. Note: When <dest> is <hwregs> and <src> is <hwregs>, <dest> cannot be EXTn if <src> is EXTn, <dest> cannot be X if <src> is X, <dest> cannot be SR if <src> is SR. Note: When <src> is <accind> <dest> cannot be A.			
MLD	Multiply	MLD<srcl>,<srcl>[,<bank switch>]	<hwregs>,<regind>	1	1	MLD A@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MLD A@P1:0,OFF
			<regind>,<regind>	1	1	MLD @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MLD @P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <band switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			
MPYA	Multiply and add	MPYA <srcl>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYA A@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYA A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYA @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYA@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			

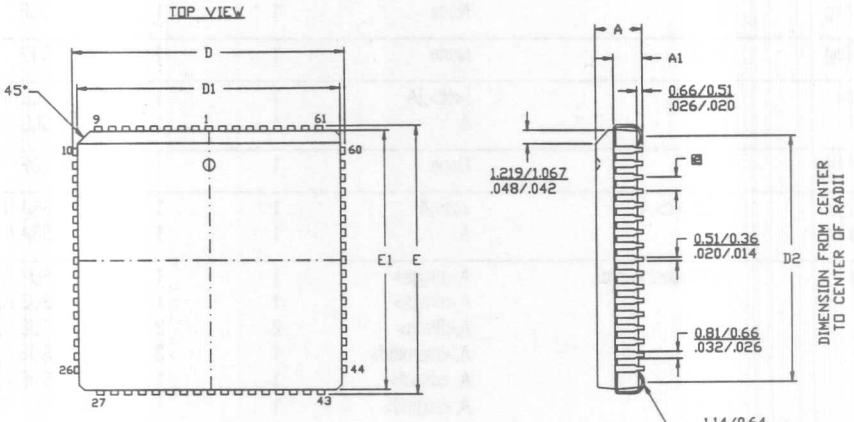
DSP INSTRUCTION DESCRIPTIONS (Continued)

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
MPYS	Multiply and subtract	MPYS<src1>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYS A,@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYS A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYS @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYS@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register.			
			Note: <hwregs> for src1 cannot be X.			
			Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, <regind> the <bank switch> defaults to ON.			
NEG	Negate	NEG <cc>,A	<cc>, A	1	1	NEG NZ,A
			A	1	1	NEG A
NOP	No operation	NOP	None	1	1	NOP
OR	Bitwise OR	OR <dest>,<src>	A, <pregs>	1	1	OR A,#128
			A, <dregs>	1	1	OR A, D0:1
			A, <limm>	2	2	OR A,@@P0:0+LOOP
			A, <memind>	1	3	OR A,@P2:1+
			A, <direct>	1	1	OR A, X
			A, <regind>	1	1	
			A, <hwregs>	1	1	
POP	Pop value from stack	POP <dest>	<pregs>	1	1	POP P0:0
			<pregs>	1	1	POP D0:1
			<regind>	1	1	POP @P0:0
			<hwregs>	1	1	POP A
						POP BUS
PUSH	Push value onto stack	PUSH <src>	<pregs>	1	1	PUSH P0:0
			<dregs>	1	1	PUSH D0:1
			<regind>	1	1	PUSH @P0:0
			<hwregs>	1	1	PUSH A
			<limm>	2	2	PUSH BUS
			<accind>	1	3	PUSH #12345
			<memind>	1	3	PUSH @A
						PUSH @@P0:0
RET	Return from subroutine	RET	None	1	2	RET
RL	Rotate Left	RL <cc>,A	<cc>,A	1	1	RL NZ,A
			A	1	1	RL A
RR	Rotate Right	RR <cc>,A	<cc>,A	1	1	RR NZ,A
			A	1	1	RR A

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
SCF	Set C flag	SCF	None	1	1	SCF
SIEF	Set IE flag	SIEF	None	1	1	SIEF
SLL	Shift left logical	SLL	[<cc>,]A	1	1	SLL NZ,A
			A	1	1	SLL A
SOPF	Set OP flag	SOPF	None	1	1	SOPF
SRA	Shift right arithmetic	SRA<cc>,A	<cc>,A	1	1	SRA NZ,A
			A	1	1	SRA A
SUB	Subtract	SUB<dest>,<src>	A,<pregs>	1	1	SUB A,#128
			A,<dregs>	1	1	SUB A,D0:1
			A,<limm>	2	2	SUB A,@@P0:0+LOOP
			A, <memind>	1	3	SUB A,@P2:1+
			A, <direct>	1	1	SUB A,X
			A, <regind>	1	1	
			A, <hwregs>	1	1	
XOR	Bitwise exclusive OR	XOR <dest>,<src>	A, <pregs>	1	1	XOR A,#128
			A, <dregs>	1	1	XOR A,D0:1
			A, <limm>	2	2	XOR A,@@P0:0+LOOP
			A, <memind>	1	3	XOR A,@P2:1+
			A, <direct>	1	1	XOR A, X
			A, <regind>	1	1	
			A, <hwregs>	1	1	

Bank Switch Enumerations. The third (optional) operand of the MLD, MPYA and MPYS instructions represents whether a bank switch is set on or off. To more clearly represent this two keywords are used (ON and OFF) which state the direction of the switch. These keywords are referred to in the instruction descriptions via the <bank switch> symbol.

PACKAGE INFORMATION



NOTES:
1. CONTROLLING DIMENSIONS : INCH
2. LEADS ARE COPLANAR WITHIN .004 IN.
3. DIMENSION : MM / INCH

SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.32	4.57	.170	.180
A1	2.67	2.92	.105	.115
D/E	25.02	25.40	.985	1.000
D1/E1	24.13	24.33	.950	.958
D2	22.86	23.62	.900	.930
e	1.27 TYP		.050 TYP	

68-Pin PLCC Package Diagram

ORDERING INFORMATION

Z89120	Z89920
20 MHz	**20 MHz**
68-Pin PLCC	68-Pin PLCC
Z8912020VSC	Z8992021VSC

Codes

Speed
20 = 20.48MHz

Package
V = Plastic Leaded Chip Carrier (PLCC)

Temperature
S = 0°C to + 70°C

Environment
C = Plastic Standard

Example:

Z 89120 10 V S C is a Z89120, 20.48 MHz, PLCC, 0°C to +70°C, Plastic Standard Flow

- C — Environmental Flow
- S — Temperature
- V — Package
- 10 — Speed
- 89120 — Product Number
- Z — Zilog Prefix

3

Z89121
Z89921 (ROMLESS)
16-BIT MIXED SIGNAL PROCESSOR

FEATURES

- Z8® Microcontroller with 43 I/O Lines (27 I/O Lines for the Z89921)
- 24 Kbytes of Z8 Program ROM (Z89121)
- 256 Bytes On-Chip Z8 RAM
- Watch-Dog Timer and Power-On Reset
- Low Power STOP Mode
- On-Chip Oscillator which Accepts a Crystal or External Clock Drive
- Two 8-Bit Z8 Counter/Timers with 6-Bit Prescaler
- Low Power Consumption - 200 mW (typical)
- Two Comparators with Programmable Interrupt Priority
- Six Vectored, Z8 Prioritized Interrupts
- RAM and ROM Protect
- Clock Speed of 20.48 MHz
- 16-Bit Digital Signal Processor (DSP)
- 6K Words DSP Program ROM
- 512 Words On-Chip DSP RAM
- 10-Bit PWM D/A Converter (4 kHz to 64 kHz)
- Z8 and DSP Operation in Parallel
- Three Vectored, Prioritized DSP Interrupts
- IBM® PC-Based Development Tools
- Interface for Two Codecs with 8 kHz and 6.66 kHz Sampling Rate and 2.048 MHz Clock
- Two DSP Timers to Support Different Sampling Rates for Codecs and PWM
- Built-in DRAM Interface. Direct Support of up to 48 Mbit DRAM with 4-Bit Wide Data Bus

4

GENERAL DESCRIPTION

The Z89121/921 is a dual CPU 16-bit mixed signal processor designed for digital audio compression plus storage systems. The I/O control processor is a Z8® with 24 Kbytes of program memory, two 8-bit counter timers, a DRAM controller with up to 48 Mbit accessibility and up to 43 /O pins. The DSP is a 16-bit processor with a 24-bit ALU and accumulator, 512 x 16 bits of RAM, single cycle instructions, and 6K word program ROM. The chip also contains a 10-bit PWM D/A converter and interface for two Codecs. The sampling rates for the PWM and Codec interface are programmable.

The Z8 and DSP processors are coupled by mailbox registers and an interrupt system. DSP or Z8 programs may be directed by events in each other's domain.

The Z89921 is the ROMless version of the Z89121. The DSP is not ROMless. The DSP's program memory is always the internal ROM.

Notes:
All Signals with a preceding front slash, "/", are active Low, e.g.: B/W (WORD is active Low); /B/W (BYTE is active Low, only).

Power connections follow conventional descriptions below:

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

GENERAL DESCRIPTION (Continued)

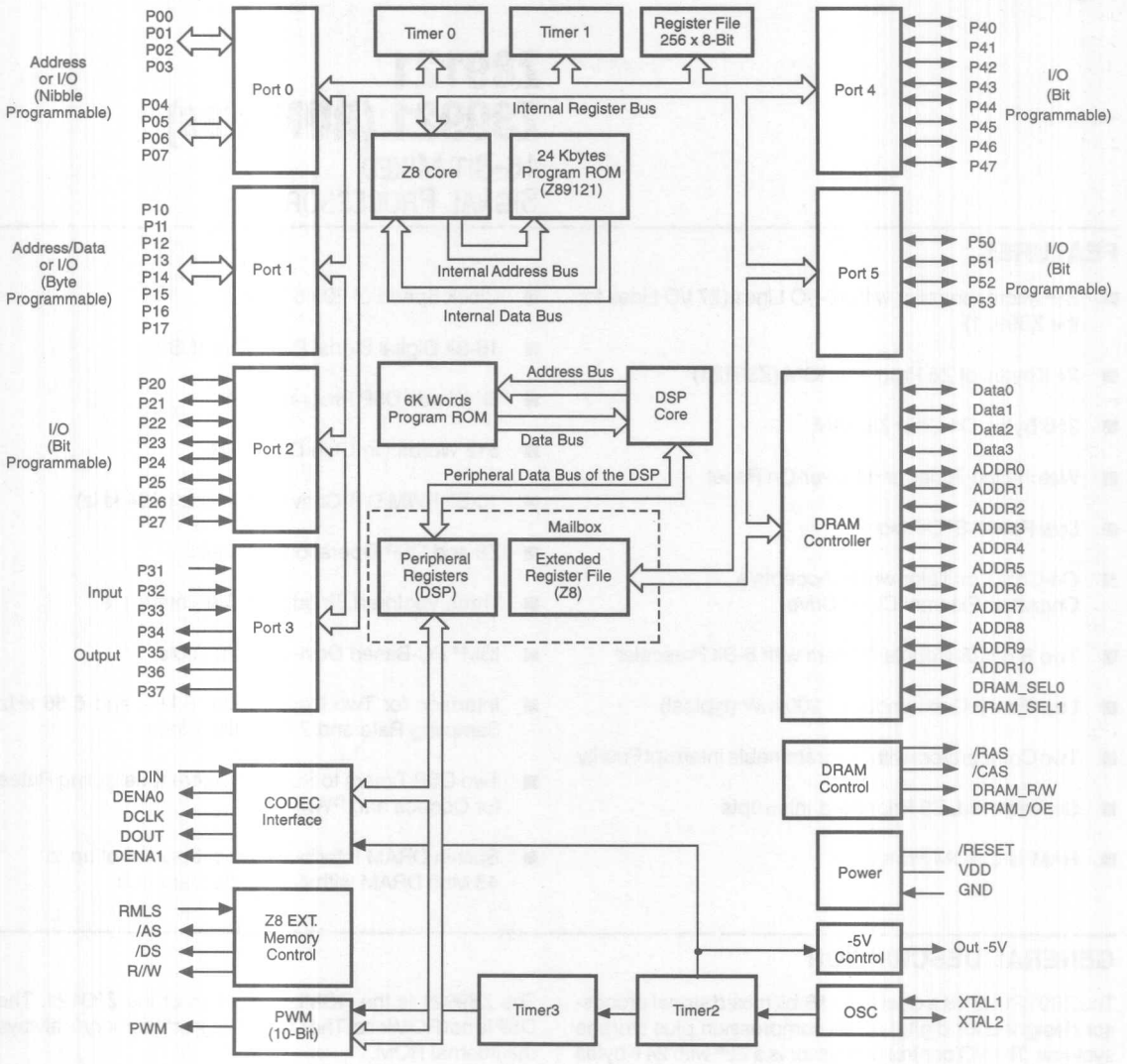


Figure 1. Functional Block Diagram

Z8 Core Processor

The Z8 is Zilog's 8-bit microcontroller core with an Expanded Register File to allow access to register-mapped peripheral and I/O circuits. The Z8 offers a flexible I/O scheme, an efficient register and address space structure, and a number of ancillary features. The Z8 also excels in many industrial uses, high-volume processing, peripheral controllers and consumer applications.

For applications demanding powerful I/O capabilities, the Z89121/921 has 43 pins dedicated to input and output. These lines are grouped into six ports. Each port is configurable under software control to provide timing, status signals and parallel I/O with or without handshake.

Four basic memory resources for the Z8 are available to support a wide range of configurations: Program Memory, Register File, Data Memory, and Expanded Register File. The Z8 core processor is characterized by an efficient register file that allows any of 256 on-board data and control registers to be the source and/or the destination of almost any instruction. Traditional microprocessor Accumulator bottlenecks are eliminated.

The Register File is composed of 236 bytes of general-purpose registers, four I/O port registers, and 15 control and status registers. The Expanded Register File consists of mailbox registers, WDT mode register, DSP Control register, Stop-Mode Recovery register, Port Configuration register, and the control and data registers for Port 4 and Port 5.

To unburden the software from supporting real-time problems, such as counting/timing and data communication, the Z8 offers two on-chip counter/timers with a large number of user selectable modes.

Watch-Dog Timer and Stop-Mode Recovery features are software driven by setting specific bits in control registers.

Stop and Halt instructions support reduced power operation. The low power-stop mode allows parameter information to be stored in the register file if power fails. An external capacitor or battery retains power to the device.

DSP Coprocessor

The DSP coprocessor is a second generation, 16-bit two's complement CMOS Digital Signal Processor (DSP). Four external DSP registers are mapped into the expanded register file of the Z8. Communication between the Z8 and the DSP occurs through those common registers which form the mailbox registers.

The analog signal is generated by a 10-bit resolution Pulse Width Modulator. The PWM output is a digital signal with CMOS output levels. The output signal has a resolution of 1 in 1024 with a sampling rate of 16 kHz (XTAL = 20.48 MHz). The sampling rate can be changed under software control and can be set at 4, 10, 16, and 64 kHz. The dynamic range of the PWM is from 0 to 4 Volts.

Two additional timers (Timer2 and Timer3) have been added to support different sampling rates for the Codec interface and Pulse Width Modulator. These timers are free-running counters that divide the crystal frequency.

PIN DESCRIPTION (Continued)

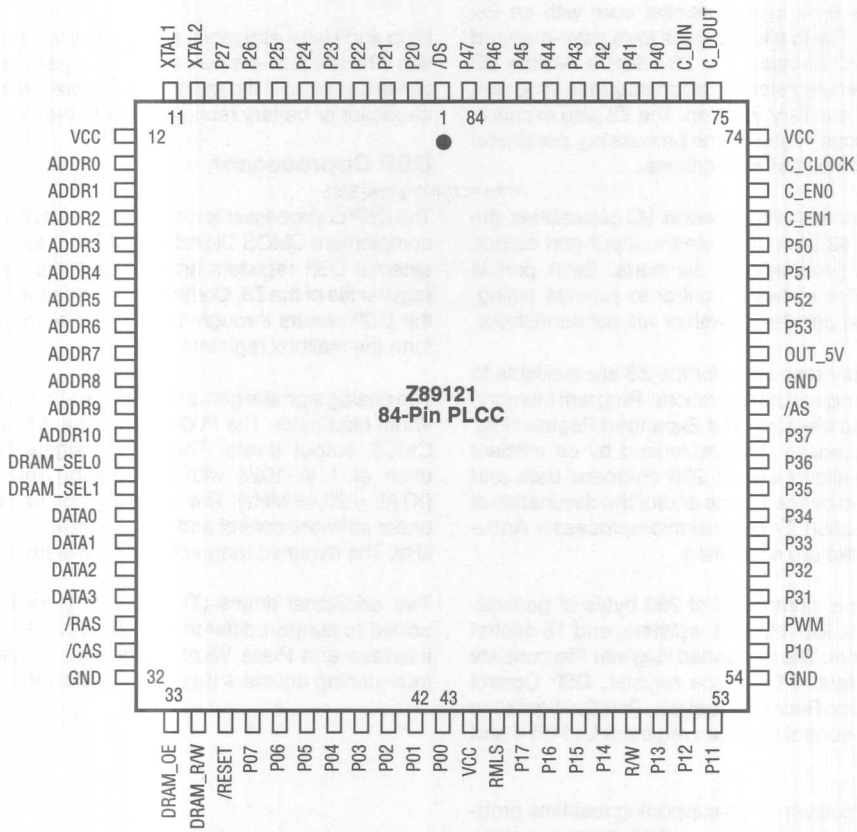


Figure 2. Z89121 84-Pin PLCC Pin Assignments

Table 1. Z89121 84-Pin Plastic Leaded Chip Carrier, Pin Identification

I/O Port Functions	Pin Number	I/O	Function
V_{SS}	32, 54, 65		Digital Ground
V_{CC}	12, 44, 74		Digital V_{CC} = +5V
P07-P00	43-36	Input/Output	P07-P00 (General purpose nibble programmable I/O port.)
P17-P10	55, 53-51, 49-46	Input/Output	P17-P10 (General purpose byte programmable I/O port.)
P27-P20	2-9	Input/Output	P27-P20 (General purpose bit programmable I/O.)
P37-P31	57-63	Input/Output	P37-P31 (General purpose I/O port. Bits P31-P33 are inputs, while bits P37-P34 are outputs.)
P47-P40	77-84	Input/Output	P47-P40 (General purpose bit programmable I/O.)
P53-P50	70-67	Input/Output	P53-P50 (General purpose bit programmable I/O.)
C_DIN	76	Input	Data input from Codec.
C_DOUT	75	Output	Data output to Codec.
C_CLOCK	73	Output	Codec clock (2.048 MHz)
C_ENA0	72	Output	Codec0 enable (8 kHz)
C_ENA1	71	Output	Codec1 enable (8 kHz)
PWM	56	Output	Pulse Width Modulator output
DATA0	26	Input/Output	Data 0 I/O of the DRAM Interface
DATA1	27	Input/Output	Data 1 I/O of the DRAM Interface
DATA2	28	Input/Output	Data 2 I/O of the DRAM Interface
DATA3	29	Input/Output	Data 3 I/O of the DRAM Interface
ADDR0	13	Output	Address 0 line of the DRAM Interface
ADDR1	14	Output	Address 1 line of the DRAM Interface
ADDR2	15	Output	Address 2 line of the DRAM Interface
ADDR3	16	Output	Address 3 line of the DRAM Interface
ADDR4	17	Output	Address 4 line of the DRAM Interface
ADDR5	18	Output	Address 5 line of the DRAM Interface
ADDR6	19	Output	Address 6 line of the DRAM Interface
ADDR7	20	Output	Address 7 line of the DRAM Interface
ADDR8	21	Output	Address 8 line of the DRAM Interface
ADDR9	22	Output	Address 9 line of the DRAM Interface
ADDR10	23	Output	Address 10 line of the RAM Interface for 4 Meg ARAMs. Select 2 output of DRAM Interface for 1 Meg ARAMs support. The latter mode is used to switch between different pages of ARAM.
DRAM_SEL0	24	Output	Select0 output of DRAM Interface. Used to switch between different pages of DRAM.
DRAM_SEL1	25	Output	Select1 output of DRAM Interface. Used to switch between different pages of DRAM.
/RAS	30	Output	Row Address Strobe of DRAM Interface.
/CAS	31	Output	Column Address Strobe of DRAM Interface.
DRAM_R/W	34	Output	Read/Write Strobe of DRAM Interface.
DRAM_/OE	33	Output	Output Enable Strobe of DRAM Interface.
XTAL1	11	Input	20.48 MHz crystal input
XTAL2	10	Output	20.48 MHz crystal output
ROMless	45	Input	Z8 ROMless mode input (P0 and P1 are switched to D/A mode if this pin is connected to V_{CC}). Internally this pin is tied to GND.
/Reset	35	Input	/RESET input
R/W	50	Output	Z8 external memory interface R/W output
/AS	64	Output	Z8 external memory interface /AS output
/DS	1	Output	Z8 external memory interface /DS
OUT_5V	66	Output	–5V charge pump

PIN DESCRIPTION (Continued)

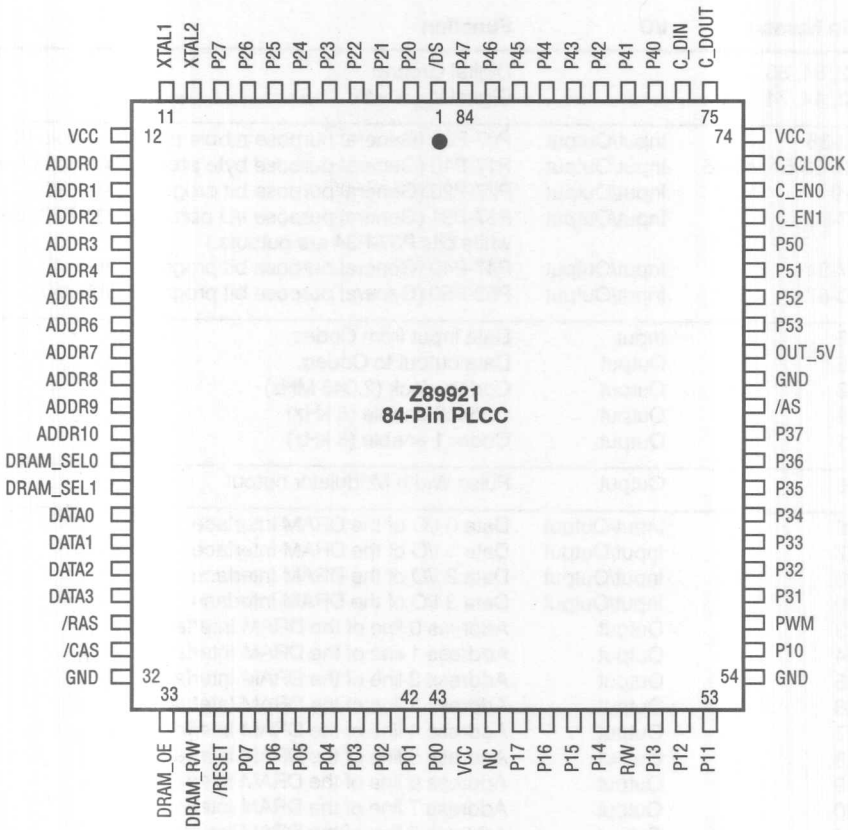


Figure 3. Z89921 84-Pin PLCC Pin Assignments

Table 2. Z89921 84-Pin Plastic Leaded Chip Carrier, Pin Identification

I/O Port Functions	Pin Number	I/O	Function
V_{SS}	32, 54, 65		Digital Ground
V_{CC}	12, 44, 74		Digital V_{CC} = +5 V
P07-P00	43-36	Input/Output	P07-P00 (General purpose nibble programmable I/O port.)
P17-P10	55, 53-51, 49-46	Input/Output	P17-P10 (General purpose byte programmable I/O port.)
P27-P20	2-9	Input/Output	P27-P20 (General purpose bit programmable I/O.)
P37-P31	57-63	Input/Output	P37-P31 (General purpose I/O port. Bits P31-P33 are inputs, while bits P37-P34 are outputs.)
P47-P40	77-84	Input/Output	P47-P40 (General purpose bit programmable I/O.)
P53-P50	70-67	Input/Output	P53-P50 (General purpose bit programmable I/O.)
C_DIN	76	Input	Data input from Codec.
C_DOUT	75	Output	Data output to Codec.
C_CLOCK	73	Output	Codec clock (2.048 MHz)
C_ENA0	72	Output	Codec0 enable (8 kHz)
C_ENA1	71	Output	Codec1 enable (8 kHz)
PWM	56	Output	Pulse Width Modulator output
DATA0	26	Input/Output	Data 0 I/O of the DRAM Interface
DATA1	27	Input/Output	Data 1 I/O of the DRAM Interface
DATA2	28	Input/Output	Data 2 I/O of the DRAM Interface
DATA3	29	Input/Output	Data 3 I/O of the DRAM Interface
ADDR0	13	Output	Address 0 line of the DRAM Interface
ADDR1	14	Output	Address 1 line of the DRAM Interface
ADDR2	15	Output	Address 2 line of the DRAM Interface
ADDR3	16	Output	Address 3 line of the DRAM Interface
ADDR4	17	Output	Address 4 line of the DRAM Interface
ADDR5	18	Output	Address 5 line of the DRAM Interface
ADDR6	19	Output	Address 6 line of the DRAM Interface
ADDR7	20	Output	Address 7 line of the DRAM Interface
ADDR8	21	Output	Address 8 line of the DRAM Interface
ADDR9	22	Output	Address 9 line of the DRAM Interface
ADDR10	23	Output	Address 10 line of the DRAM Interface for 4 Meg ARAMs. Select 2 output of DRAM Interface for 1 Meg ARAMs support. The latter mode is used to switch between different pages of ARAM.
DRAM_SEL0	24	Output	Select 0 output of DRAM Interface. Used to switch between different pages of DRAM.
DRAM_SEL1	25	Output	Select 1 output of DRAM Interface. Used to switch between different pages of DRAM.
/RAS	30	Output	Row Address Strobe of DRAM Interface.
/CAS	31	Output	Column Address Strobe of DRAM Interface.
DRAM_R/W	34	Output	Read/Write Strobe of DRAM Interface.
DRAM_/OE	33	Output	Output Enable Strobe of DRAM Interface.
XTAL1	11	Input	20.48 MHz crystal input
XTAL2	10	Output	20.48 MHz crystal output
NC	45	Not Connected	
/Reset	35	Input	/RESET input
R/W	50	Output	Z8 external memory interface R/W output
/AS	64	Output	Z8 external memory interface /AS output
/DS	1	Output	Z8 external memory interface /DS output
OUT_5V	66	Output	–5V Charge Pump

PIN FUNCTIONS

/RESET (input, active Low). Initializes the MCU. Reset is accomplished either through Power-On Reset (POR), Watch-Dog Timer reset, Stop-Mode Recovery, or external reset. During POR and WDT Reset, the internally generated reset signal is driving the reset pin Low for the POR time. Any devices driving the reset line must be open drain to avoid damage from a possible conflict during reset conditions. A /RESET will reset both the Z8 and the DSP.

For the Z8:
After the POR time, /RESET is a Schmitt-triggered input. To avoid asynchronous and noisy reset problems, the Z8 is equipped with a reset filter of four external clocks (4TpC). If the external reset signal is less than 4TpC in duration, no reset occurs. On the fifth clock after the reset is detected, an internal RST signal is latched and held for an internal register count of 18 external clocks, or for the duration of the external reset, whichever is longer. Program execution begins at location 000CH (Hexadecimal), 5-10 TpC cycles after the /RESET is released. The Z8 does not reset WDT, SMR, P2M, and P3M registers on a Stop-Mode Recovery operation.

For the DSP:
A low level on the /RESET pin generates an internal reset signal. The /RESET signal must be kept low for at least one clock cycle. The CPU will fetch a new Program Counter (PC) value from program memory address 0FFCH after the reset signal is released.

ROMless (input, active High). This pin, when connected to V_{DD}, disables the internal Z8 ROM. (Note, when pulled Low to GND the part functions normally as the ROM version.) The DSP can not be configured as ROMless. This pin is available only on the Z89121.

R/W *Read/Write* (output, write Low). The R/W signal defines the signal flow when the Z8 is reading or writing to external program or data memory. The Z8 is reading when this pin is High and writing when this pin is Low.

/AS *Address Strobe* (output, active Low). Address Strobe is pulsed once at the beginning of each machine cycle. Address output is via Port 0/Port 1 for all external programs. Memory address transfers are valid at the trailing edge of /AS. Under program control, /AS is placed in the high-impedance state along with Ports 0 and 1, Data Strobe, and Read/Write.

/DS *Data Strobe* (output, active Low). Data Strobe is activated once for each external memory transfer. For read operations, data must be available prior to the trailing edge of /DS. For write operations, the falling edge of /DS indicates that output data is valid.

XTAL1 *Crystal 1* (time-based input). This pin connects a parallel-resonant crystal, ceramic resonator, LC, RC network or an external single-phase clock to the on-chip oscillator input.

XTAL2 *Crystal 2* (time-based output). This pin connects a parallel-resonant, crystal, ceramic resonant, or LC network to the on-chip oscillator output.

PWM *Pulse Width Modulator* (output). The PWM is a 10-bit resolution D/A converter. This output is a digital signal with CMOS output levels.

V_{DD}. Digital power supply for the Z89121/921.

GND. Digital ground for the Z89121/921.

C_DIN (input). Data input from Codec.

C_DOUT (output). Data output to Codec.

C_CLOCK (output). 2.048 MHz data rate clock signal output to Codec.

C_ENA0 (output). Enable signal to Codec0

C_ENA1 (output). Enable signal to Codec1.

DRAM_SEL0 (output). Select0 of DRAM.

DRAM_SEL1 (output). Select1 of DRAM.

Port 0 (P07-P00). Port 0 is an 8-bit, bidirectional, CMOS compatible port. These eight I/O lines are configured under software control as a nibble I/O port, or as an address port for interfacing external memory. The input buffers are Schmitt-triggered and the output drivers are push-pull. Port 0 is placed under handshake control. In this configuration, Port 3, lines P32 and P35 are used as the handshake control /DAV0 and RDY0. Handshake signal direction is dictated by the I/O direction to Port 0 of the upper nibble P07-P04. The lower nibble must have the same direction as the upper nibble.

The Auto Latch on Port 0 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1 cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

For external memory references, Port 0 provides address bits A11-A8 (lower nibble) or A15-A8 (lower and upper nibble) depending on the required address space. If the address range requires 12 bits or less, the upper nibble of Port 0 can be programmed independently as I/O while the lower nibble is used for addressing. If one or both nibbles are needed for I/O operation, they are configured by writing to the Port 0 mode register.

In ROMless mode, after a hardware reset, Port 0 is configured as address lines A15-A8, and extended timing is set to accommodate slow memory access. The initialization routine can include reconfiguration to eliminate this extended timing mode. (In ROM mode, Port 0 is defined as input after reset.)

Port 0 is set in the high-impedance mode if selected as an address output state along with Port 1 and the control signals /AS, /DS and R//W (Figure 4).

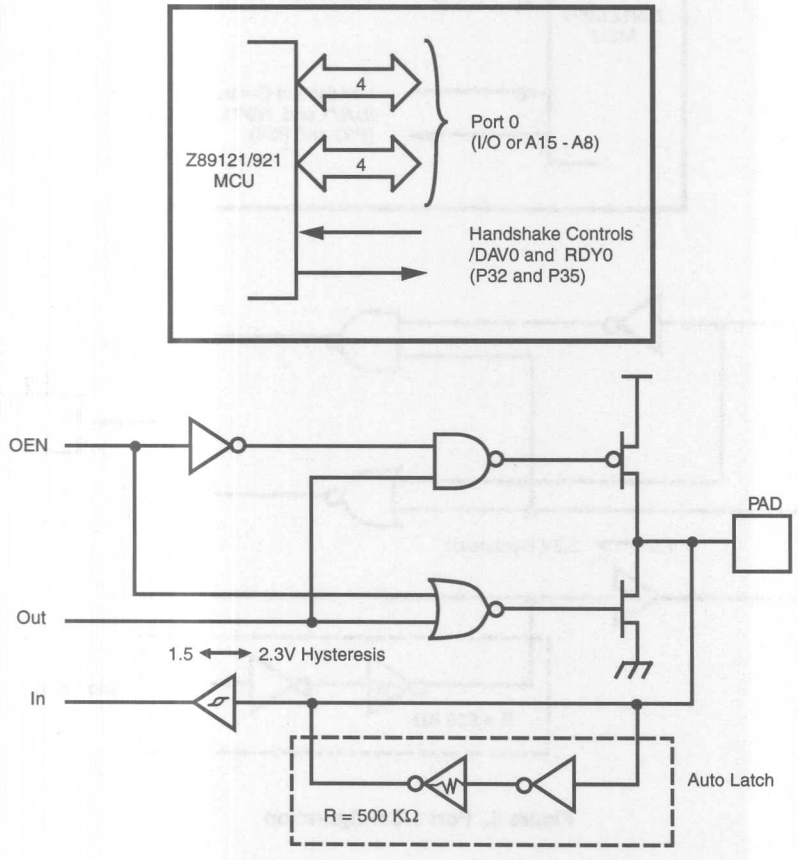


Figure 4. Port 0 Configuration

PIN FUNCTIONS (Continued)

Port 1 (P17-P10). Port 1 is an 8-bit, bidirectional, CMOS compatible port (Figure 5). It has multiplexed Address (A7-A0) and Data (D7-D0) ports. These eight I/O lines are programmed as inputs or outputs, or can be configured under software control as an Address/Data port for interfacing external memory. The input buffers are Schmitt-triggered and the output drivers are push-pull.

Port 1 may be placed under handshake control. In this configuration, Port 3, lines P33 and P34 are used as the handshake controls RDY1 and /DAV1 (Ready and Data Available). Memory locations greater than 24575 (in ROM mode) are referenced through Port 1. To interface external memory, Port 1 must be programmed for the multiplexed Address/Data mode. If more than 256 external locations are required, Port 0 outputs the additional lines.

Port 1 can be placed in the high-impedance state along with Port 0, /AS, /DS and R/W, allowing the Z89121/68 to share common resources in multiprocessor and DMA applications.

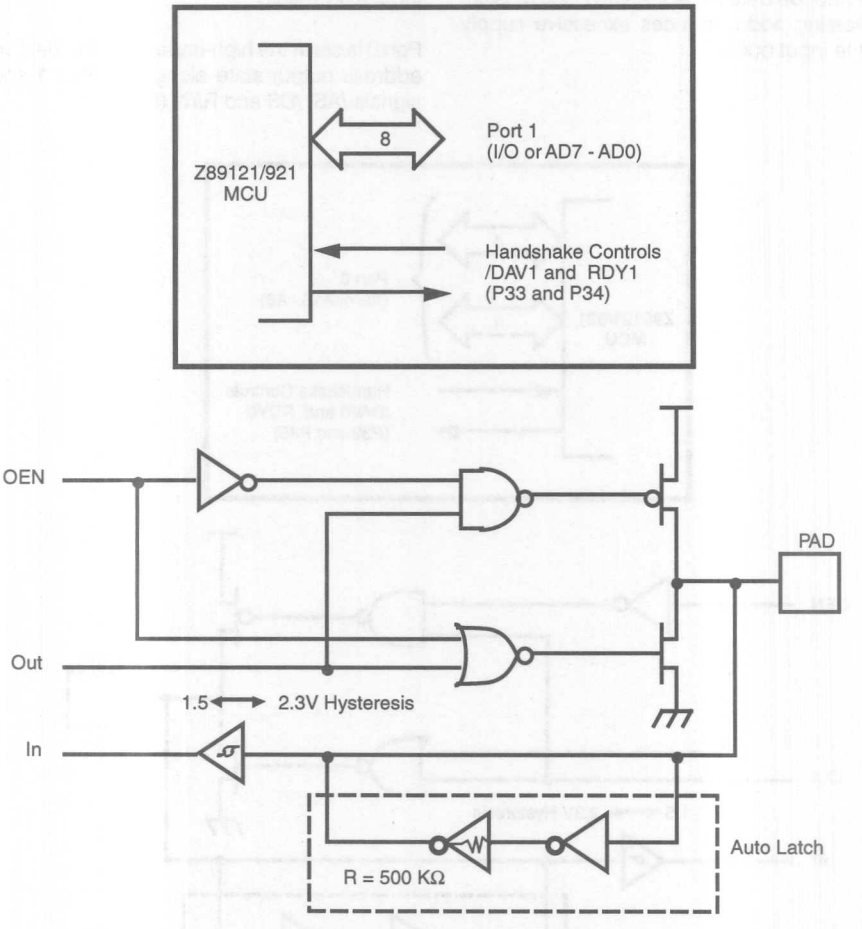


Figure 5. Port 1 Configuration

Port 2. (P27-P20.) Port 2 is an 8-bit, bidirectional, CMOS compatible I/O port. These eight I/O lines are configured under software control independently as inputs or outputs. Port 2 is always available for I/O operation. The input buffers are Schmitt-triggered. Bits programmed as outputs may be globally programmed as either push-pull or open-drain.

Port 2 may be placed under handshake control. In this configuration, Port 3 lines P31 and P36 are used as the handshake controls lines /DAV2 and RDY2. The handshake signal assignment for Port 3 lines P31 and P36 is dictated by the direction (input or output) assigned to bit 7, Port 2 (Figure 6).

The Auto Latch on Port 2 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1, cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

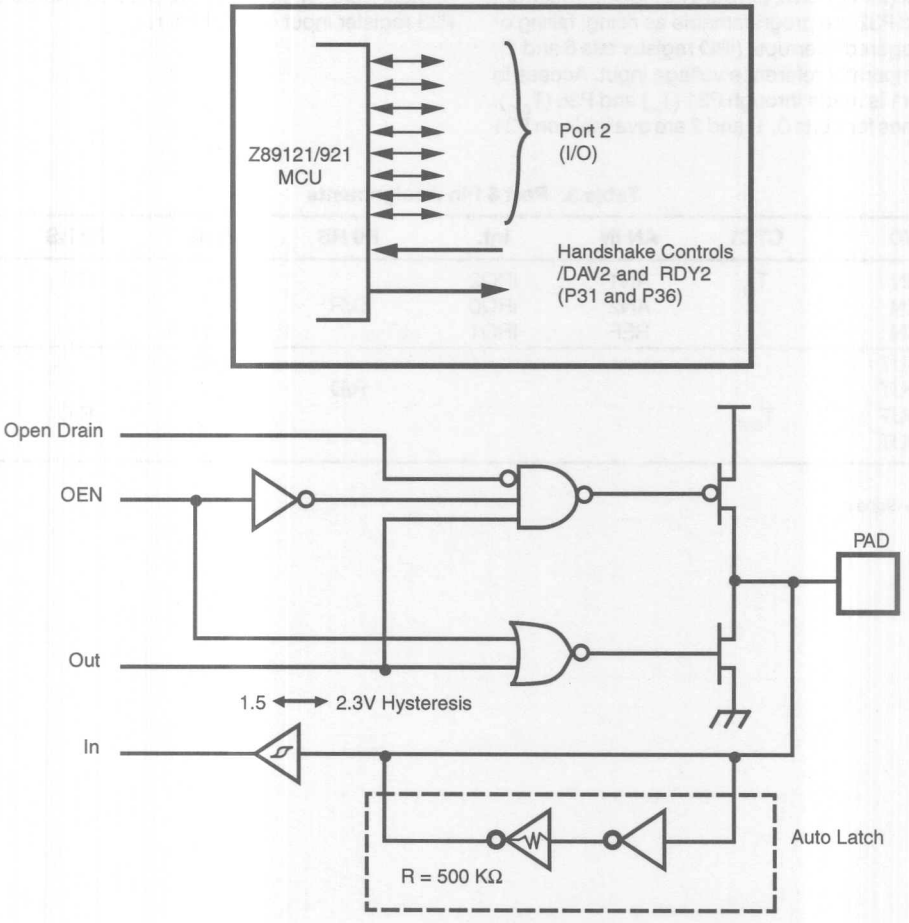


Figure 6. Port 2 Configuration

4

PIN FUNCTIONS (Continued)

Port 3 (P37-P31). Port 3 is a 7-bit, CMOS compatible port with three fixed inputs (P33-P31) and four fixed outputs (P37-P34). It is configured under software control for input/output, counter/timers, interrupt, and port handshakes. Pins P33, P32, and P31 are standard CMOS inputs; outputs are push-pull.

Two on-board comparators can process analog signals on P31 and P32 with reference to the voltage on P33. The analog function is enabled by programming the Port 3 Mode Register (bit 1). Port 3, pin 3 is a falling edge interrupt input. P31 and P32 are programmable as rising, falling or both edge-triggered interrupts (IRQ register bits 6 and 7). P33 is the comparator reference voltage input. Access to Counter/Timer1 is made through P31 (T_{IN}) and P36 (T_{OUT}). Handshake lines for ports 0, 1, and 2 are available on P31 through P36.

Port 3 also provides the following control functions: handshake for Ports 0, 1, and 2 (/DAV and RDY); three external interrupt request signals (IRQ3-IRQ1); timer input and output signals (T_{IN} and T_{OUT}); (Figure 7).

Comparator Inputs. Port 3, pins P31 and P32 both have a comparator front end. The comparator reference voltage, pin P33, is common to both comparators. In analog mode, P31 and P32 are the positive inputs to the comparators and P33 is the reference voltage supplied to both comparators. In digital mode, pin P33 can be used as a P33 register input or IRQ1 source.

Table 3. Port 3 Pin Assignments

Pin	I/O	CTC1	AN IN	Int.	P0 HS	P1HS	P2 HS	EXT
P31	IN	T_{IN}	AN1	IRQ2			D/R	
P32	IN		AN2	IRQ0	D/R			
P33	IN		REF	IRQ1		D/R		
P34	OUT					R/D		DM
P35	OUT				R/D			
P36	OUT	T_{OUT}					R/D	
P37	OUT							

Notes:
HS = Handshake Signals
D = DAV
R = RDY

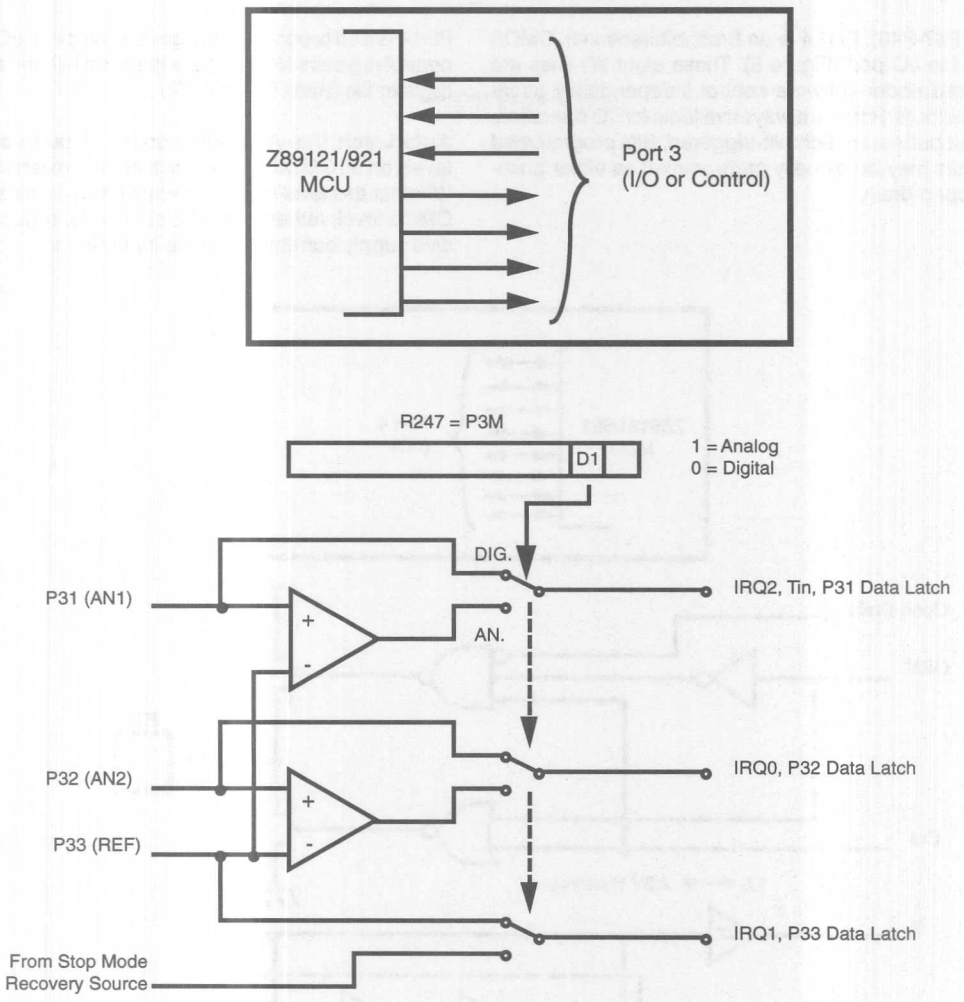


Figure 7. Port 3 Configuration

4

PIN FUNCTIONS (Continued)

Port 4 (P47-P40). Port 4 is an 8-bit, bidirectional, CMOS compatible I/O port (Figure 8). These eight I/O lines are configured under software control independently as inputs or outputs. Port 4 is always available for I/O operation. The input buffers are Schmitt-triggered. Bits programmed as outputs may be globally programmed as either push-pull or open-drain.

Port 4 is a bit programmable general purpose I/O port. The control registers for Port 4 are mapped into the expanded register file (Bank F) of the Z8.

Auto Latch. The Auto Latch on Port 4 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1 cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

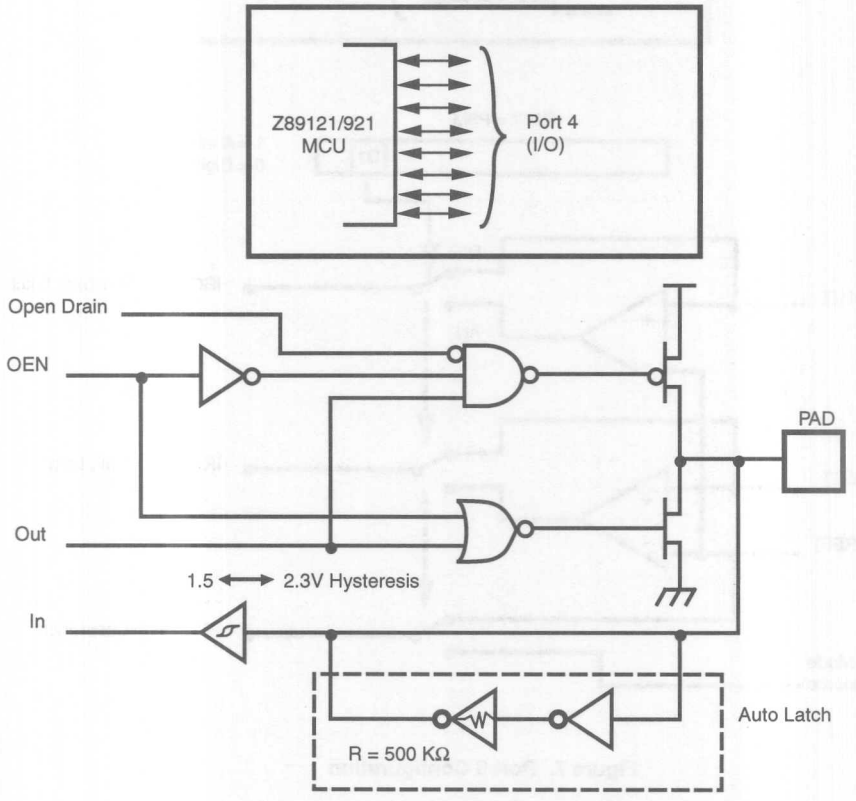


Figure 8. Port 4 Configuration

Port 5 (P53-P50). Port 5 is an 4-bit, bidirectional, CMOS compatible I/O port (Figure 9). These four I/O lines are configured under software control independently as inputs or outputs. Port 5 is always available for I/O operation. The input buffers are Schmitt-triggered. Bits programmed as outputs may be globally programmed as either push-pull or open-drain.

Port 5 is a bit programmable general purpose I/O port. The control registers for Port 5 are mapped into the expanded register file (Bank F) of the Z8.

Auto Latch. The Auto Latch on Port 5 puts valid CMOS levels on all CMOS inputs which are not externally driven. Whether this level is 0 or 1 cannot be determined. A valid CMOS level, rather than a floating node, reduces excessive supply current flow in the input buffer.

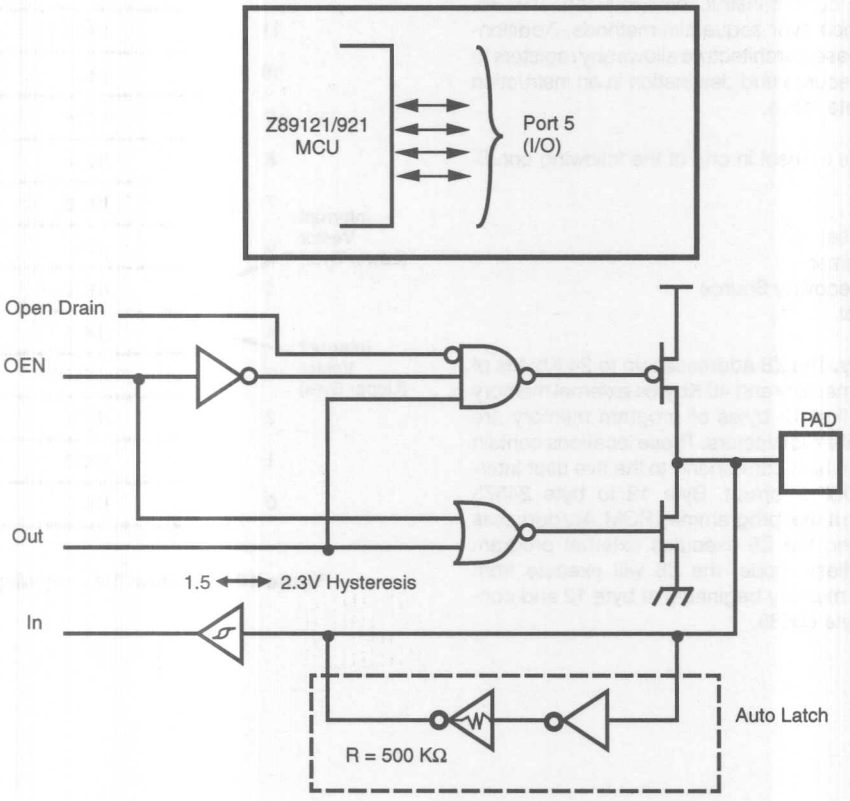


Figure 9. Port 5 Configuration

4

Z8® FUNCTIONAL DESCRIPTION

The Z8 CCP core incorporates special functions to enhance the Z8's performance in control applications.

Pipelined Instructions. The Z8 instructions (see page 66) are comprised of two parts, an instruction fetch and execute part. The instructions typically take between six and ten cycles to fetch and five cycles to execute. Five cycles of the next instruction fetch may be overlapped with five cycles of the current instruction execution. This improves performance over sequential methods. Additionally, the register-based architecture allows any registers to be picked as the source and destination in an instruction saving intermediate move.

Reset. The device is reset in one of the following conditions:

- Power-On Reset
- Watch-Dog Timer
- Stop-Mode Recovery Source
- External Reset

Program Memory. The Z8 addresses up to 24 Kbytes of internal program memory and 40 Kbytes external memory (Figure 10). The first 12 bytes of program memory are reserved for the interrupt vectors. These locations contain six 16-bit vectors which correspond to the five user interrupts and one DSP interrupt. Byte 12 to byte 24575 consists of on-chip mask-programmed ROM. At addresses 24576 and greater, the Z8 executes external program memory. In ROMless mode, the Z8 will execute from external program memory beginning at byte 12 and continuing through byte 65535.

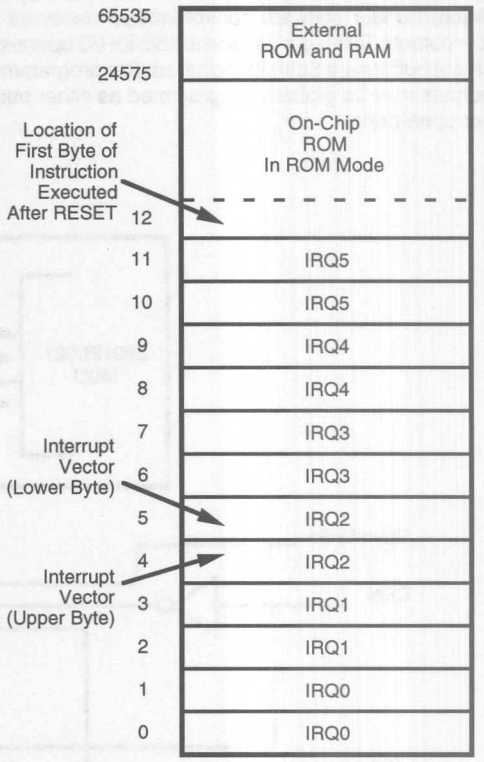


Figure 10. Program Memory Map

ROM Protect. The 24 Kbyte of internal program memory for the Z8 is mask programmable. A ROM protect feature prevents "dumping" of the ROM contents of Program Memory by inhibiting execution of LDC, LDCI, LDE, and LDEI instructions. The ROM Protect option is mask-programmable, to be selected by the customer at the time when the ROM code is submitted.

Data Memory (/DM). In ROM mode, the Z8 can address up to 40 Kbytes of external data memory beginning at location 24576 (Figure 11). In ROMless mode, the Z8 can address the full 64 Kbytes of external data memory beginning at location 12. External data memory may be included with, or separated from, the external program memory space. /DM, an optional I/O function that can be programmed to appear on Port 34, is used to distinguish between data and program memory space (Table 3). The state of the /DM signal is controlled by the type of instruction being executed. An LDC opcode references program (/DM inactive) memory, and an LDE instruction references data (/DM active Low) memory.

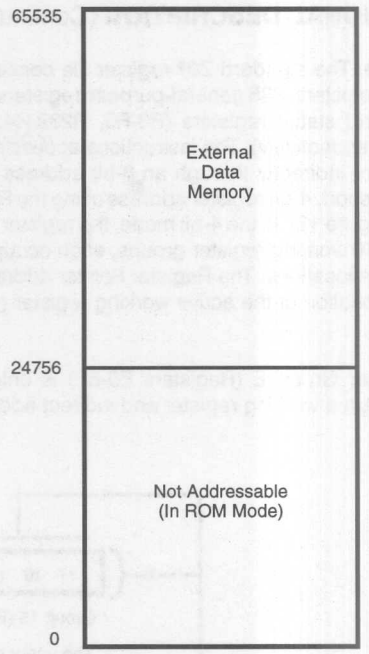


Figure 11. Data Memory Map

Z8 FUNCTIONAL DESCRIPTION (Continued)

Register File. The standard Z8® register file consists of four I/O port registers, 236 general-purpose registers, and 15 control and status registers (R3-R0, R239-R4, and R255-R241, respectively). The instructions access registers directly or indirectly through an 8-bit address field. This allows a short, 4-bit register address using the Register Pointer (Figure 12). In the 4-bit mode, the register file is divided into 16 working register groups, each occupying 16 continuous locations. The Register Pointer addresses the starting location of the active working register group (Figure 13).

Note: Register Group E (Registers E0-EF) is only accessed through a working register and indirect addressing modes.

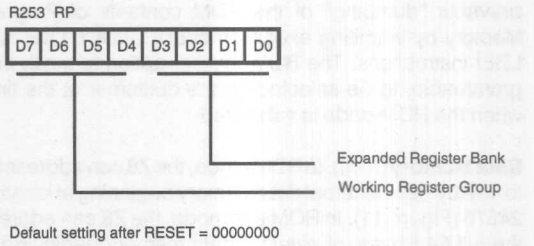


Figure 12. Register Pointer Register

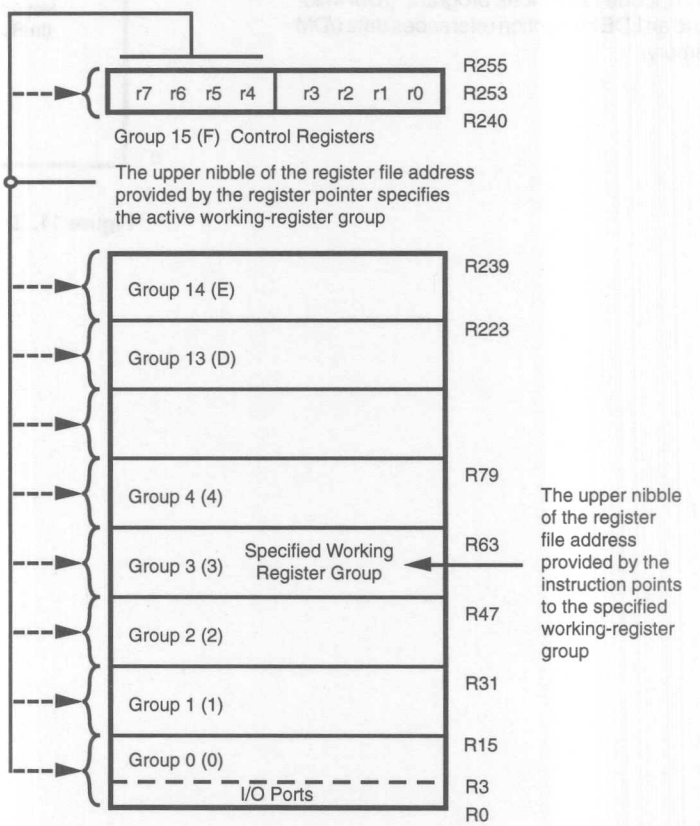


Figure 13. Register Pointer

RAM Protect. The upper portion of the Z8's RAM address spaces 90H to EFH (excluding the control registers) is protected from reading and writing. The RAM Protect bit option is mask-programmable and is selected by the customer when the ROM code is submitted. After the mask option is selected, the user activates the RAM Protect from the internal ROM code by loading bit D6 in the IMR register to either a 0 (off) or a 1 (on). A 1 in D6 indicates RAM Protect enabled.

Stack. The Z8's external data memory or the internal register file is used for the stack. The 16-bit Stack Pointer (R255-R254) is used for the external stack which can reside only from 24576 to 65535 in ROM mode or 0 to 65535 in ROMless mode. An 8-bit Stack Pointer (R255) is used for the internal stack that resides within the 236 general-purpose registers (R239-R4). SPH can be used as a general-purpose register when using internal stack only.

Expanded Register File. The register file on the Z8 has been expanded to allow for additional system control registers, and for mapping of additional peripheral devices, along with I/O ports, into the register address area. The Z8 register address space has now been implemented as 16 banks of 16 register groups per bank (Figure 14). These register banks are known as the ERF (Expanded Register File). Bits 7-4 of register RP (Register Pointer) select the working register group. Bits 3-0 of register RP select the expanded register bank (Figure 14).

The SMR register, WDT Register, control and data registers for Port 4 and Port 5, and the DSP control register are located in Bank F of the Expanded Register File. Bank B of the Expanded Register File consists of the Mailbox Interface in which the Z8 and the DSP communicate. The rest of the Expanded Register is not physically implemented and is open for future expansion.

Z8 FUNCTIONAL DESCRIPTION (Continued)

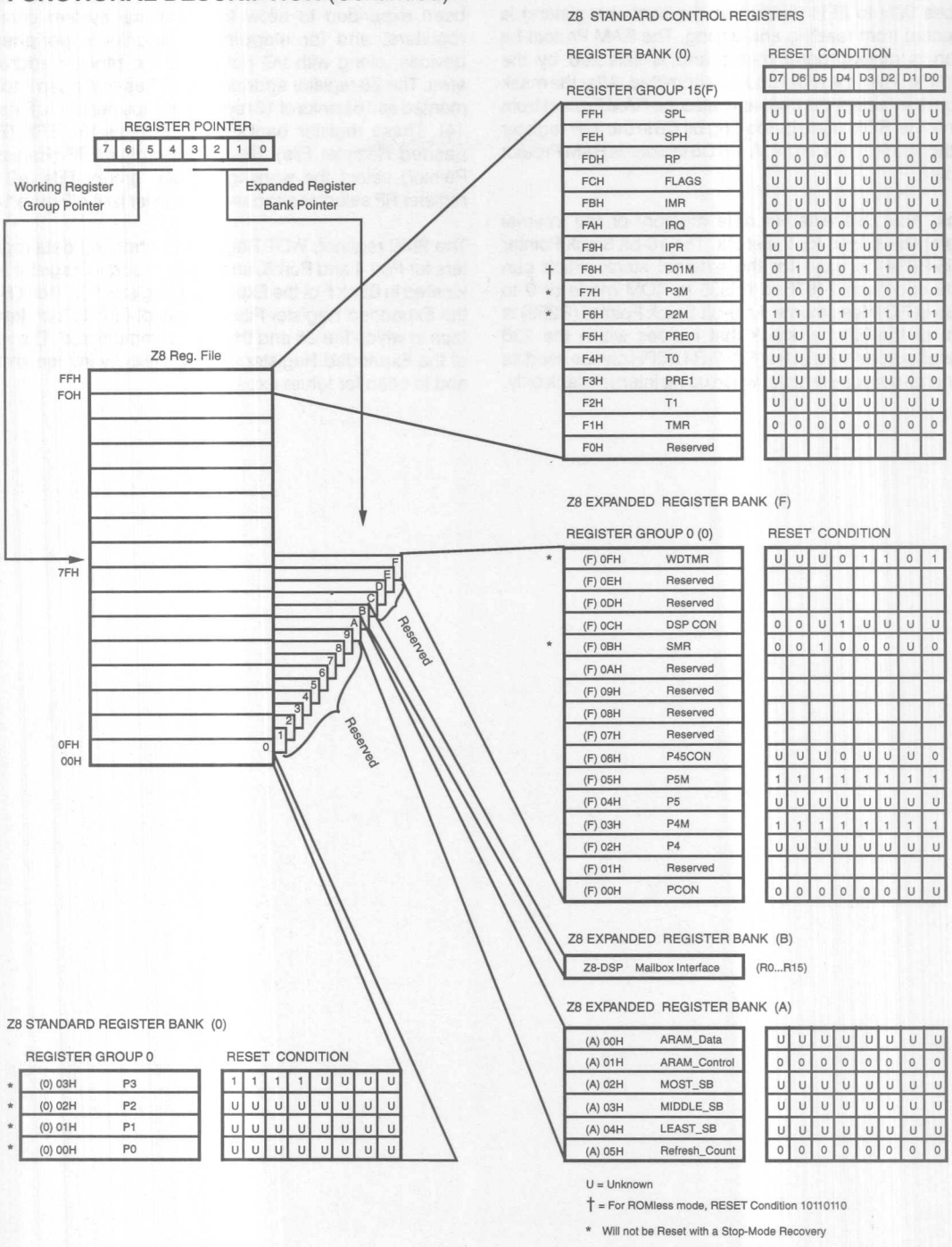


Figure 14. Expanded Register File Architecture

Interrupts. The Z8 has six different interrupts from six different sources. The interrupts are maskable and prioritized (Figure 15). The six sources are divided as follows; three sources are claimed by Port 3 lines P33-P31, two by counter/timers, and one by the DSP (Table 4). The Interrupt Mask Register globally or individually enables or disables the six interrupt requests.

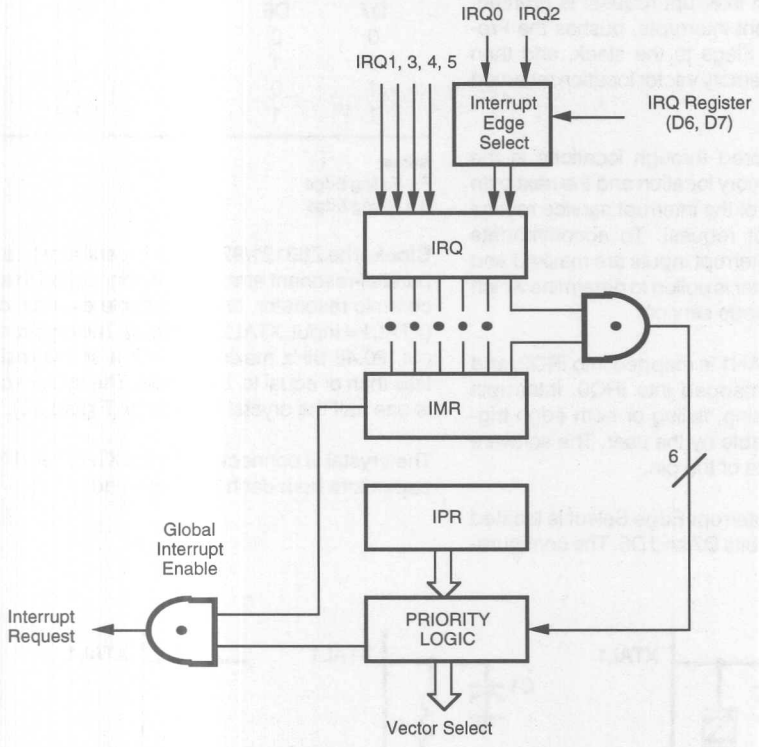


Figure 15. Interrupt Block Diagram

Table 4. Interrupt Types, Sources, and Vectors

Name	Source	Vector Location	Comments
IRQ0	/DAV0, P32	0, 1	External (P32), Programmable Rise or Fall Edge Triggered
IRQ1	/DAV1, P33	2, 3	External (P33), Fall Edge Triggered
IRQ2	/DAV2, P31, T_{IN}	4, 5	External (P31), Programmable Rise or Fall Edge Triggered
IRQ3	IRQ3	6, 7	Internal (DSP activated), Fall Edge Triggered
IRQ4	T0	8, 9	Internal
IRQ5	TI	10, 11	Internal

Z8 FUNCTIONAL DESCRIPTION (Continued)

When more than one interrupt is pending, priorities are resolved by a programmable priority encoder controlled by the Interrupt Priority Register. An interrupt machine cycle is activated when an interrupt request is granted. This disables all subsequent interrupts, pushes the Program Counter and Status Flags to the stack, and then branches to the program memory vector location reserved for that interrupt.

All Z8 interrupts are vectored through locations in the program memory. This memory location and the next byte contain the 16-bit address of the interrupt service routine for that particular interrupt request. To accommodate polled interrupt systems, interrupt inputs are masked and the Interrupt Request Register is polled to determine which of the interrupt requests needs service.

An interrupt resulting from AN1 is mapped into IRQ2, and an interrupt from AN2 is mapped into IRQ0. Interrupts IRQ2 and IRQ0 may be rising, falling or both edge triggered, and are programmable by the user. The software may poll to identify the state of the pin.

Programming bits for the Interrupt Edge Select is located in the IRQ Register (R250), bits D7 and D6. The configuration is shown in Table 5.

Table 5. IRQ Register

IRQ		Interrupt Edge	
D7	D6	P31	P32
0	0	F	F
0	1	F	R
1	0	R	F
1	1	R/F	R/F

Notes:
F = Falling Edge
R = Rising Edge

Clock. The Z89121/921 on-chip oscillator has a high-gain, parallel-resonant amplifier for connection to a crystal, LC, ceramic resonator, or any suitable external clock source (XTAL1 = Input, XTAL2 = Output). The crystal should be AT cut, 20.48 MHz maximum, with a series resistance (RS) less than or equal to 100 Ohms. The system clock (SCLK) is one half the crystal frequency (Figure 16).

The crystal is connected across XTAL1 and XTAL2 using capacitors from each pin to ground.

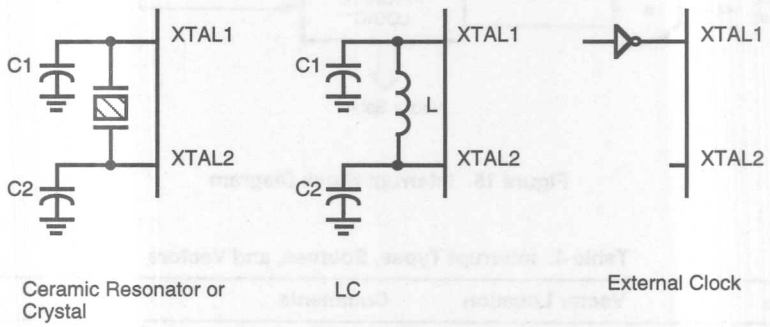


Figure 16. Oscillator Configuration

Counter/Timers. There are two 8-bit programmable counter/timers (T0-T1), each driven by its own 6-bit programmable prescaler. The T1 prescaler is driven by internal or external clock sources; however, the T0 prescaler is driven by the internal clock only (Figure 17).

The 6-bit prescalers can divide the input frequency of the clock source by any integer number from 1 to 64. Each prescaler drives its counter, which decrements the value (1 to 256) that has been loaded into the counter. When the counter reaches the end of the count, a timer interrupt request, IRQ4 (T0) or IRQ5 (T1), is generated.

The counters can be programmed to start, stop, restart to continue, or restart from the initial value. The counters can also be programmed to stop upon reaching zero (single pass mode) or to automatically reload the initial value and continue counting (modulo-n continuous mode).

The counters, but not the prescalers, are read at any time without disturbing their value or count mode. The clock source for T1 is user-definable and is either the internal microprocessor clock divided by four, or an external signal input via Port 31. The Timer Mode register configures the external timer input (P31) as an external clock, a trigger input that can be retriggerable or non-retriggerable, or as a gate input for the internal clock. The counter/timers can be cascaded by connecting the T0 output to the input of T1.

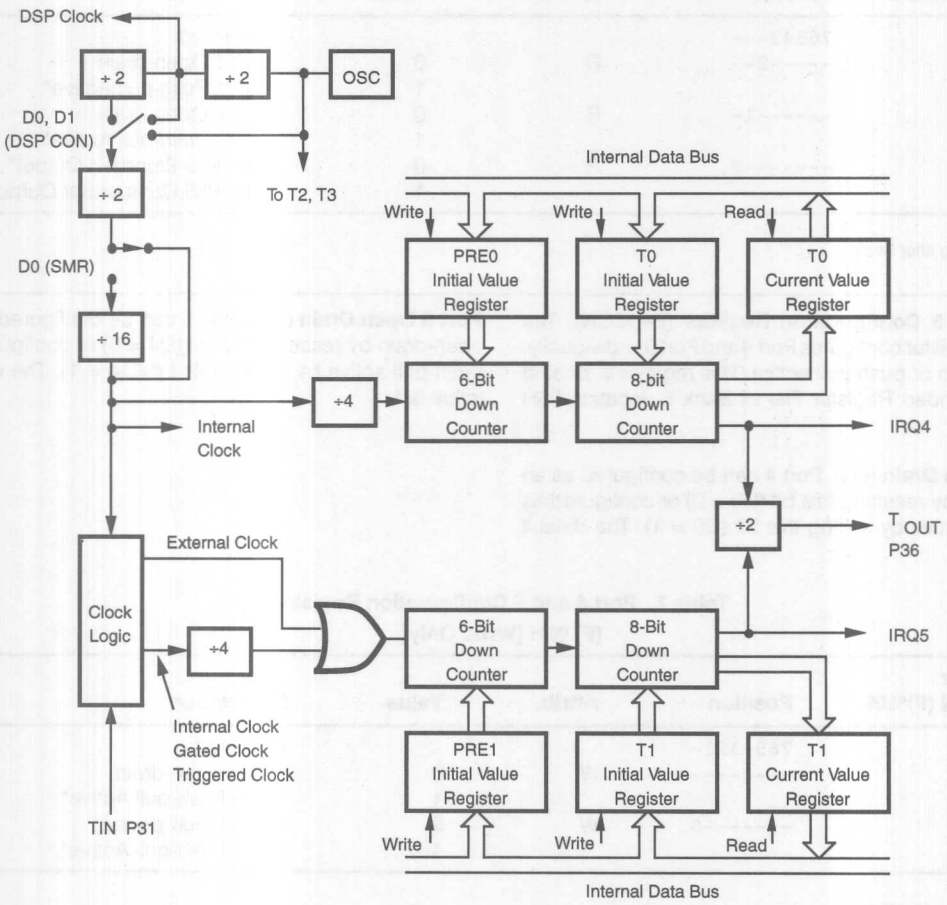


Figure 17. Counter/Timer Block Diagram

Z8 FUNCTIONAL DESCRIPTION (Continued)

Port Configuration Register (PCON). The PCON register configures each port individually; comparator output on Port 3, and open-drain on Port 0 and Port 1. The PCON register is located in the Expanded Register File at bank F, location 00H (Table 6).

Comparator Output Port 3 (D0). Bit 0 controls the comparator use in Port 3. A 1 in this location brings the comparator outputs to P34 and P35, and a 0 releases the Port to its standard I/O configuration.

Port 0 Open Drain (D1). Port 0 can be configured as an open-drain by resetting this bit (D1 = 0) or configured as push-pull active by setting this bit (D1 = 1). The default value is 1.

Port 1 Open Drain (D2). Port 1 can be configured as an open-drain by resetting this bit (D2 = 0) or configured as push-pull active by setting this bit (D2 = 1). The default value is 1.

Table 6. Port Configuration Register (PCON) (F) 00H

Register PCON (F)%00	Position	Attrib.	Value	Description
	76543---			Reserved
	-----2--	R	0	Port 1 Open-drain
			1	Port 1 Push-pull Active*
	------1-	R	0	Port 0 Open-drain
			1	Port 0 Push-pull Active*
	-------0	R	0	P34, P35 Standard Output*
			1	P34, P35 Comparator Output

Note:
* Default setting after Reset

Port 4 and 5 Configuration Register (P45CON). The P45CON register configures Port 4 and Port 5, individually, to open-drain or push-pull active. This register is located in the Expanded Register File at Bank F, location 06H (Table 7).

Port 4 Open Drain (D0). Port 4 can be configured as an open-drain by resetting this bit (D0 = 0) or configured as push-pull active by setting this bit (D0 = 1). The default value is 1.

Port 5 Open Drain (D4). Port 5 can be configured as an open-drain by resetting this bit (D4 = 0) or configured as push-pull active by setting this bit (D4 = 1). The default value is 1.

Table 7. Port 4 and 5 Configuration Register (F) 06H [Write Only]

Register P45CON (F)%06	Position	Attrib.	Value	Description
	765-321-			Reserved
	---4----	W	0	Port 5 Open-drain
			1	Port 5 Push-pull Active*
	-------0	W	0	Port 4 Open-drain
			1	Port 4 Push-pull Active*

Note:
* Default setting after Reset

Power-On Reset (POR). A timer circuit clocked by a dedicated on-board RC oscillator is used for the Power-On Reset (POR) timer function. The POR time allows V_{CC} and the oscillator circuit to stabilize before instruction execution begins.

The POR timer circuit is a one-shot timer triggered by one of three conditions:

1. Power fail to Power OK status.
2. Stop-Mode Recovery (if D5 of SMR = 1).
3. WDT time-out.

The POR time is a nominal 5 ms. Bit 5 of the Stop-Mode Register determines whether the POR timer is bypassed after Stop-Mode Recovery (typical for external clock, RC/LC oscillators).

HALT. HALT turns off the internal CPU clock, but not the XTAL oscillation. The counter/timers and external interrupts IRQ0, IRQ1, IRQ2, and IRQ3 remain active. The devices are recovered by interrupts, either externally or internally generated.

STOP. This instruction turns off the internal clock and external crystal oscillation. It reduces the standby current to 10 µA or less. The STOP mode is terminated by a reset only, either by WDT time-out, POR, SMR recovery or external reset. This causes the processor to restart the application program at address 000CH. In order to enter STOP (or HALT) mode, it is necessary to first flush the instruction pipeline to avoid suspending execution in mid-instruction. To do this, the user must execute a NOP (opcode = FFH) immediately before the appropriate sleep instruction, i.e.;

```
FF  NOP    ; clear the pipeline
6F  STOP   ; enter Stop mode
             or
FF  NOP    ; clear the pipeline
7F  HALT   ; enter Halt mode
```

Stop-Mode Recovery Register (SMR). This register selects the clock divide value and determines the mode of Stop-Mode Recovery (Table 8). All bits are write only except bit 7, which is read only. Bit 7 is a flag bit that is hardware set on the condition of STOP recovery and reset by a power-on cycle. Bit 6 controls whether a low level or a high level is required from the recovery source. Bit 5 controls the reset delay after recovery. Bits 2, 3, and 4, or the SMR register, specify the source of the Stop-Mode Recovery signal. Bits 0 and 1 determine the time-out period of the WDT. The SMR is located in Bank F of the Expanded Register Group at address 0BH.

Table 8. Stop-Mode Recovery Register (SMR) (F) 0BH

Register SMR (F)%0B	Position	Attrib.	Value	Description
	7-------	R	0	POR*
			1	Stop Recovery
	-6------	W	0	Low Stop Recovery Level*
			1	High Stop Recovery Level
	--5-----	W	0	Stop Delay On*
			1	Stop Delay Off
	---432--	W		Stop-Mode Recovery Source
			000	POR Only*
			001	Reserved
			010	P31
			011	P32
			100	P33
			101	P27
			110	P2 NOR 0-3
			111	P2 NOR 0-7
	------1-			Reserved
	-------0	W	0	SCLK/TCLK Not Divide by 16†
			1	SCLK/TCLK Divide by 16

Notes:
* Default setting after Reset
† Reset after Stop-Mode Recovery

counter/timers and interrupt logic).

SCLK/TCLK divide-by-16 Select (D0). D0 of the SMR controls a divide-by-16 prescaler of SCLK/TCLK. The purpose of this control is to selectively reduce device power consumption during normal processor execution (SCLK control) and/or HALT mode (where TCLK sources

Stop-Mode Recovery Source (D2, D3, and D4). These three bits of the SMR specify the wake-up source of the STOP recovery (Figure 18 and Table 9).

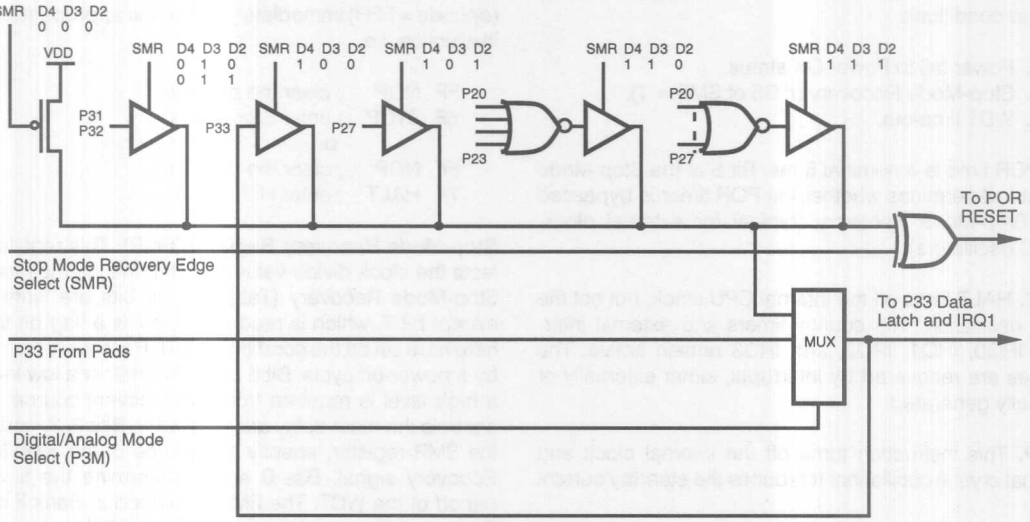


Figure 18. Stop-Mode Recovery Source

(Table 9).

Table 9. Stop-Mode Recovery Source

SMR:432			Operation
D4	D3	D2	Description of Action
0	0	0	POR and/or external reset recovery
0	0	1	Reserved
0	1	0	P31 transition
0	1	1	P32 transition
1	0	0	P33 transition
1	0	1	P27 transition
1	1	0	Logical NOR of P20 through P23
1	1	1	Logical NOR of P20 through P27

Stop-Mode Recovery Delay Select (D5). This bit, if High, disables the 5 ms /RESET delay after Stop-Mode Recovery. The default configuration of this bit is one. If the "fast" wake up is selected, the Stop-Mode Recovery source is kept active for at least five TpC.

Stop-Mode Recovery Edge Select (D6). A 1 in this bit position indicates that a high level on any one of the recovery sources wakes the Z89121/921 from STOP mode. A 0 indicates low level recovery. The default is 0 on POR

Cold or Warm Start (D7). This bit is set by the device upon entering STOP mode. It is active High, and is 0 (cold) on POR/WDT /RESET. This bit is read only. It is used to distinguish between cold or warm start.

DSP Control Register (DSPCON). The DSPCON register controls various aspects of the Z8 and the DSP. It can configure the internal system clock (SCLK) or the Z8, RESET, and HALT of the DSP, and control the interrupt interface between the Z8 and the DSP (Table 10).

Z8 IRQ3 (D0). This bit, which causes the Z8 interrupt, can be set by the DSP by writing bit 9 of ICR. Z8 has to set this bit after serving the IRQ3 interrupt. The DSP can poll the status of IRQ3 by reading ICR bit 9.

DSP INT2 (D1). This bit is linked to DSP interrupt (INT2). It can be set by the Z8. After serving INT2, the DSP has to write a 1 to an appropriate bit in ICR (EXT4) to clear the IRQ. Reading this bit reflects the status of INT2 of the DSP.

Table 10. DSP Control Register
(F) 0CH [Read/Write]

Field DSPCON (F)0CH	Position	Attrib.	Value	Label
Z8_SCLK	76------	R/W	00	2.5 MHz (OSC/8)
			01	5 MHz (OSC/4)
			1x	10 MHz (OSC/2)
DSP_Reset	--5-----	R		Return '0'
		W	0	No effect
			1	Reset DSP
DSP_Run	---4----	R/W	0	Halt_DSP
			1	Run_DSP
Reserved	----32--		xx	
				Return '0'
				No effect
IntFeedback	------1-	R		FB_DSP_INT2
		W	1	Set DSP_INT2
			0	No effect
	-------0	R		FB_Z8_IRQ3
		W	1	Clear IRQ3
			0	No effect

DSP RUN (D4). This bit defines the Halt mode of the DSP. If this bit is set to 0, then the DSP clock is turned off to minimize power consumption. After this bit is set to 1, then the DSP will continue code execution from where it was halted. After a hardware reset, this bit is reset to 1.

DSP RESET (D5). Setting this bit to 1 will reset the DSP. If the DSP was in HALT mode, this bit is automatically preset to 1. Writing a 0 has no effect.

Z8 SLCK (D6-D7). These bits define the SCLK frequency of the Z8. The oscillator can be either divided by 8, 4, or 2. After a reset, both of these are defaulted to 00.

Watch-Dog Timer Mode Register (WDTMR). The WDT is a retriggerable one-shot timer that resets the Z8 if it reaches its terminal count. The WDT is initially enabled by executing the WDT instruction and refreshed on subsequent executions of the WDT instruction. The WDT circuit is driven by an on-board RC oscillator or external oscillator from the XTAL1 pin. The POR clock source is selected with bit 4 of the WDT register (Table 11).

4

Table 11. Watch-Dog Timer Mode Register (F) 0F

Register WDTMR (F)%0F	Position	Attrib	Value	Description
	765-----			Reserved
	---4----	R/W	0	On-Board RC for WDT*
			1	XTAL for WDT
	----3---	R/W	0	WDT Off During STOP
			1	WDT On During STOP*
	-----2--	R/W	0	WDT Off During HALT
			1	WDT On During HALT*
	------10	R/W		Int RC Osc Ext. Clock
			00	5 ms 256 TpC
			01	15 ms 512 TpC*
			10	25 ms 1024 TpC
			11	100 ms 4096 TpC

Note:
* Default setting after Reset

Z8 FUNCTIONAL DESCRIPTION (Continued)

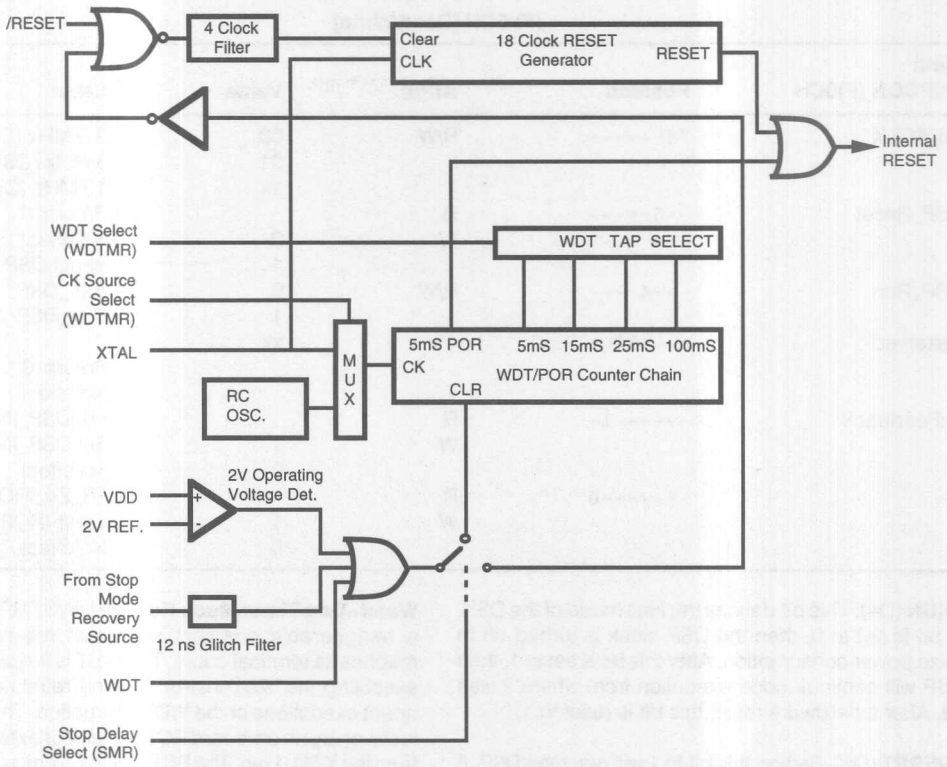


Figure 19. Resets and WDT

WDT Time Select (D0, D1). Selects the WDT time period. It is configured as shown in Table 12.

Table 12. WDT Time Select

D1	D0	Time-out of Internal RC OSC	Time-out of XTAL clock
0	0	5 ms min	256 TpC
0	1	15 ms min	512 TpC
1	0	25 ms min	1024 TpC
1	1	100 ms min	4096 TpC

Notes:
TpC = XTAL clock cycle
The default on reset is 15 ms.

WDT During HALT (D2). This bit determines whether or not the WDT is active during HALT mode. A 1 indicates active during HALT. The default is 1.

WDT During STOP (D3). This bit determines whether or not the WDT is active during STOP mode. Since XTAL clock is stopped during STOP mode, the on-board RC has to be selected as the clock source to the POR counter. A 1 indicates active during STOP. The default is 1.

Clock Source for WDT (D4). This bit determines which oscillator source is used to clock the internal POR and WDT counter chain. If the bit is a 1, the internal RC oscillator is bypassed and the POR and WDT clock source is driven from the external pin, XTAL1. The default configuration of this bit is 0 which selects the RC oscillator.

DSP FUNCTIONAL DESCRIPTION

The DSP coprocessor is characterized by an efficient hardware architecture that allows fast arithmetic operations such as multiplication, addition, subtraction and multiply accumulate of two 16-bit operands. Most instructions are executed in one clock cycle.

Four DSP registers (EXT3-EXT0) are shared through a quasi dual port mapping with the expanded register file of the Z8. Communication between the Z8 and the DSP occurs through these mailbox registers and inter-processor interrupt mechanism.

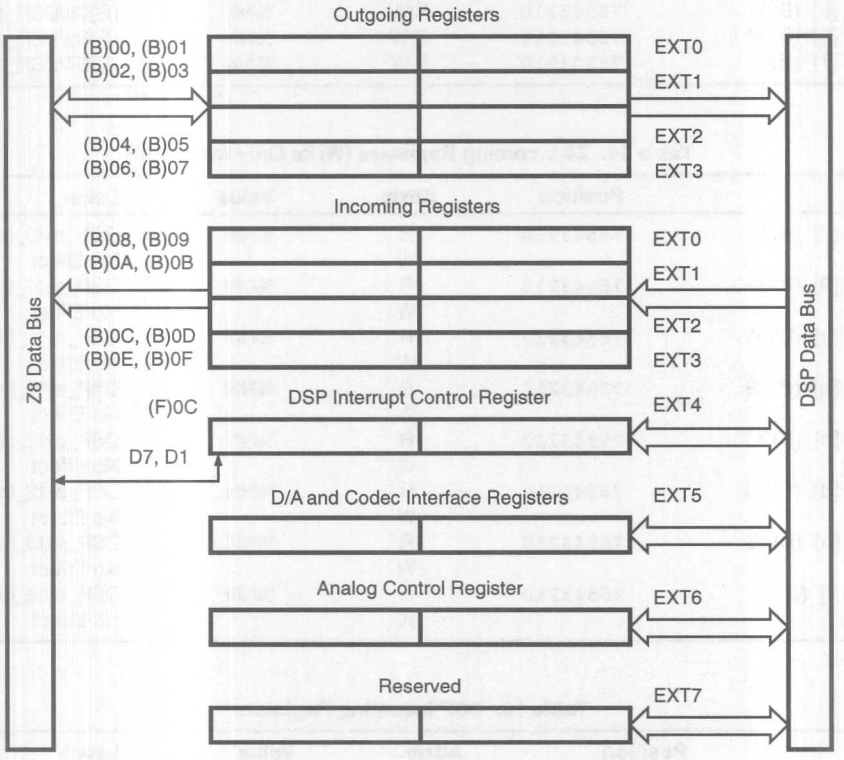


Figure 20. Z8-DSP Interface

DSP-Z8 Mail Box

To receive information from the DSP, the Z8 uses eight incoming registers which are mapped in the Z8 extended Register File (Bank B, 08 to 0F). The DSP treats these as four 16 bit registers that correspond to the eight incoming Z8 registers (Figure 20).

The Z8 can supply the DSP with data through eight outgoing registers mapped into both the Z8 Expanded Register File (Bank B, Registers 00 to 07) and the external register interface of the DSP. These registers are R/W and can be used as general purpose registers of the Z8. The DSP can only read information from these registers. Since the DSP uses a 16-bit data format and the Z8 an 8-bit data format, eight outgoing registers of the Z8 correspond to four DSP registers. The DSP can only read information from the outgoing registers.

Both the outgoing registers and the incoming registers share the same DSP address (EXT3-EXT0).

Note: The Z8 can read and write to ERF Bank B R00-R07, Registers 08-0F are read only from the Z8.

Table 13. Z8 Outgoing Registers (Read Only from DSP)

Field	Position	Attrib.	Value	Label
Outgoing [0] (B)00	76543210	R/W	%NN	(B)00/DSP_ext0_hi
Outgoing [1] (B)01	76543210	R/W	%NN	(B)01/DSP_ext0_lo
Outgoing [2] (B)02	76543210	R/W	%NN	(B)02/DSP_ext1_hi
Outgoing [3] (B)03	76543210	R/W	%NN	(B)03/DSP_ext1_lo
Outgoing [4] (B)04	76543210	R/W	%NN	(B)04/DSP_ext2_hi
Outgoing [5] (B)05	76543210	R/W	%NN	(B)05/DSP_ext2_lo
Outgoing [6] (B)06	76543210	R/W	%NN	(B)06/DSP_ext3_hi
Outgoing [7] (B)07	76543210	R/W	%NN	(B)07/DSP_ext3_lo

Table 14. Z8 Incoming Registers (Write Only from DSP)

Field	Position	Attrib.	Value	Label
Incoming [8] (B)08	76543210	R	%NN	DSP_ext0_hi
		W		No Effect
Incoming [9] (B)09	76543210	R	%NN	DSP_ext0_lo
		W		No Effect
Incoming [a] (B)0A	76543210	R	%NN	DSP_ext1_hi
		W		No Effect
Incoming [b] (B)0B	76543210	R	%NN	DSP_ext1_lo
		W		No Effect
Incoming [c] (B)0C	76543210	R	%NN	DSP_ext2_hi
		W		No Effect
Incoming [d] (B)0D	76543210	R	%NN	DSP_ext2_lo
		W		No Effect
Incoming [e] (B)0E	76543210	R	%NN	DSP_ext3_hi
		W		No Effect
Incoming [f] (B)0F	76543210	R	%NN	DSP_ext3_lo
		W		No Effect

Table 15. DSP Incoming Registers

Field	Position	Attrib.	Value	Label
DSP_ext0	fedcba9876543210	R	%NNNN	(B)00, (B)01
Mail Box		W		(B)08, (B)09
DSP_ext1	fedcba9876543210	R	%NNNN	(B)02, (B)03
Mail Box		W		(B)0A, (B)0B
DSP_ext2	fedcba9876543210	R	%NNNN	(B)04, (B)05
Mail Box		W		(B)0C, (B)0D
DSP_ext3	fedcba9876543210	R	%NNNN	(B)06, (B)07
Mail Box		W		(B)0E, (B)0F

DSP Interrupts

The DSP processor has three interrupt sources (INT2, INT1, INT0) (Figure 21). These sources have different priority levels (Figure 22). The highest priority, the next lower and the lowest priority level are assigned to INT2, INT1 and INT0, respectively. The DSP does not allow interrupt nesting (interrupting service routines that are currently being executed). When two interrupt requests occur simultaneously the DSP starts servicing the interrupt with the highest priority level.

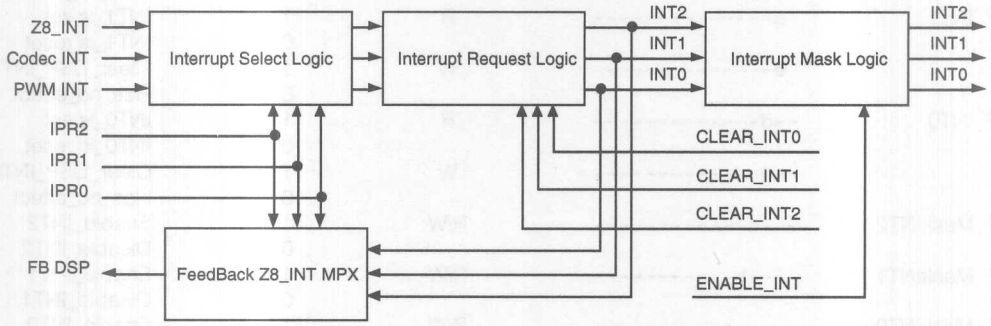


Figure 21. DSP Interrupts

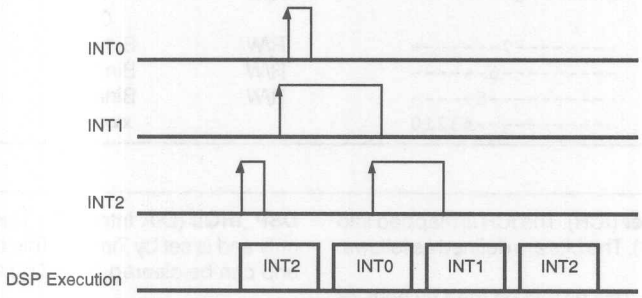


Figure 22. DSP Interrupt Priority Structure

4

DSP Interrupts (Continued)

Table 16. EXT4 DSP Interrupt Control Register (ICR) Definition

Field	Position	Attrib	Value	Label
DSP_INT2	f----------	R	1	INT2_is set
			0	INT2_is reset
	f----------	W	1	Clear_DSP_INT2
			0	Has_no_effect
DSP_INT1	-e---------	R	1	INT1_is set
			0	INT1_is reset
	-e---------	W	1	Clear_DSP_INT1
			0	Has_no_effect
DSP_INT0	--d--------	R	1	INT0_is set
			0	INT0_is reset
	--d--------	W	1	Clear_DSP_INT0
			0	Has_no_effect
DSP_MaskINT2	---c-------	R/W	1	Enable_INT2
			0	Disable_INT2
DSP_MaskINT1	----b------	R/W	1	Enable_INT1
			0	Disable_INT1
DSP_MaskINT0	-----a-----	R/W	1	Enable_INT0
			0	Disable_INT0
Z8_IRQ3	------9----	R	1	IRQ3_active
			0	IRQ3_inactive
	------9----	W	1	Set_Z8_IRQ3
			0	Has_no_effect
Enable_INT	-------8---	R/W	1	Enable_INT
			0	Disable_INT
DSP_INTSel2	--------7--	R/W	Binary	INTSel2
DSP_INTSel1	---------6-	R/W	Binary	INTSel1
DSP_INTSel0	----------5	R/W	Binary	INTSel0
Reserved	----------43210		xxxxx	Reserved

Interrupt Control Register (ICR). The ICR is mapped into EXT4 of the DSP (Table 16). The bits are defined as follows.

DSP_IRQ2 (Z8 Interrupt). This bit can be read by both Z8 and DSP and can be set only by writing to the Z8 expanded Register File (Bank F, ROC, bit 0). This bit asserts IRQ2 of the DSP and can be cleared by writing to the Clear_IRQ2 bit.

DSP_IRQ1 (A/D Interrupt). This bit can be read by the DSP only and is set when valid data is present at the A/D output register (conversion done). This bit asserts IRQ1 of the DSP and can be cleared by writing to the Clear_IRQ1bit.

DSP_IRQ0 (D/A Interrupt). This bit can be read by DSP only and is set by Timer3. This bit assists IRQ0 of the DSP and can be cleared by writing to the Clear_IRQ0 bit.

DSP_MaskIntX. These bits can be accessed by the DSP only. Writing a 1 to these locations allows the INT to be serviced, while writing a 0 masks the corresponding INT off.

Z8_IRQ3. This bit can be read from both Z8 and DSP and can be set by DSP only. Addressing this location accesses bit D3 of the Z8 IRQ register; hence, this bit is not implemented in the ICR. During the interrupt service routine executed on the Z8 side, the User has to reset the Z8_IRQ3 bit by writing a 1 to bit D0 of the DSPCON. Three Z8 instructions after this operation, the hardware of the Z89121/921 automatically resets Z8_IRQ3. This delay provides the timing synchronization between the Z8 and the DSP sides during interrupts. In summary, the interrupt service routine of the Z8 for IRQ3 should be finished by:

```
PUSH   RP
LD     RP,#%0F
OR     r12,#%01
POP    RP
IRET
```

DSP Enable_INT. Writing a 1 to this location enables global interrupts of the DSP while writing 0 disables them. A system Reset globally disables all interrupts.

DSP_IPRX. This 3-bit group defines the Interrupt Select logic according to Table 17.

Clear_IRQX. These bits can be accessed by the DSP only. Writing a 1 to these locations rests the corresponding DSP_IRQX bits to 0. Clear_IRQX are virtual bits and are not implemented.

Table 17. DSP Interrupt Selection

DSP_IPR[2-0] 2 1 0	Z8_INT is Switched to	Codec_INT is Switched to	D/A_INT is Switched to
0 0 0	INT2	INT1	INT0
0 0 1	INT1	INT2	INT0
0 1 0	INT2	INT0	INT1
0 1 1	INT1	INT0	INT2
1 0 0	INT0	INT2	INT1
1 0 1	INT0	INT1	INT2
1 1 0	Reserved	Reserved	Reserved
1 1 1	Reserved	Reserved	Reserved

4

PULSE WIDTH MODULATOR (PWM)

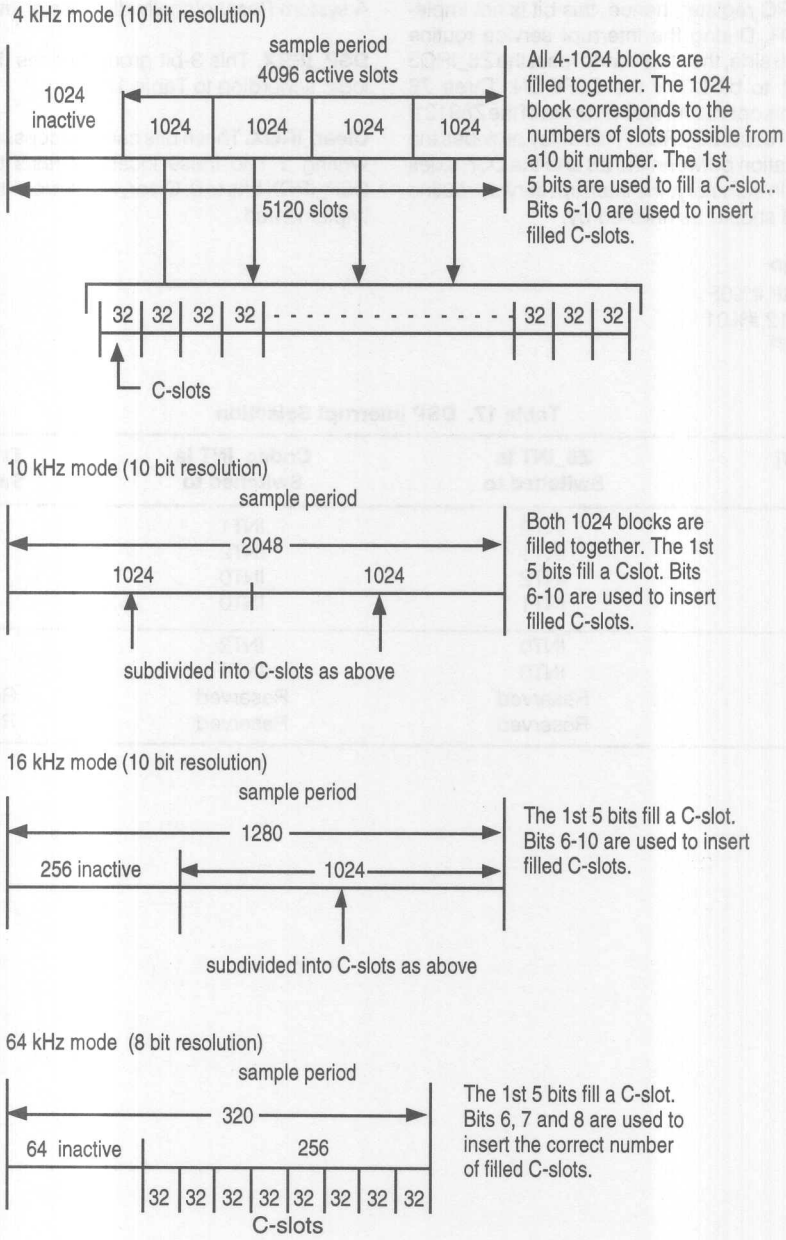


Figure 23. PWM Output

CODEC INTERFACE

Codec interface provides the user all the necessary signals to connect two independent codec chips. The supported sampling rate is 8K samples/sec. at a data rate of 2.048 MHz, or 6.66K samples/sec. at a 1.7066 MHz data rate. Figure 24 shows the connection of T2 (TCM29C18) and Motorola (MC145503) Codec to Z89121. The timing diagram is shown in Figure 25.

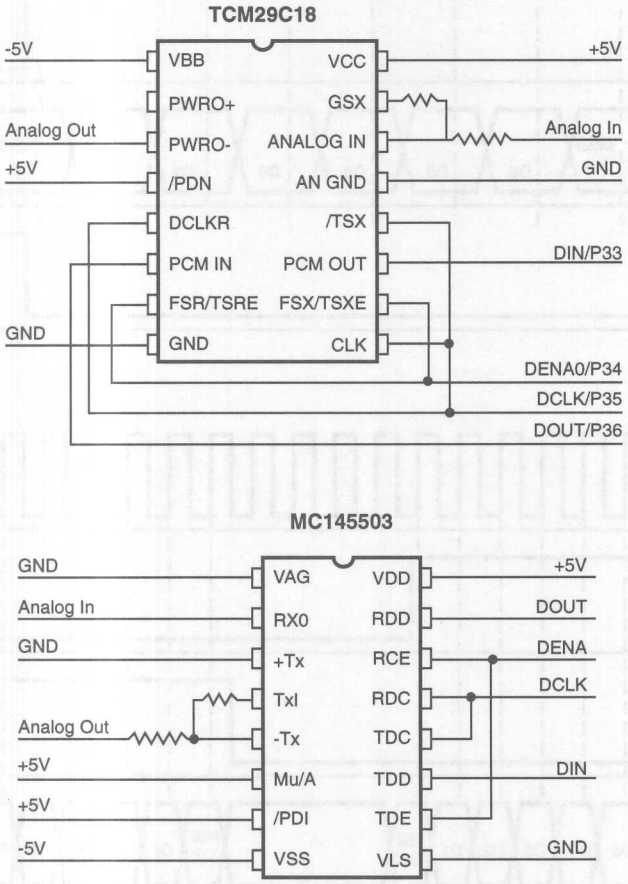


Figure 24. Connecting TCM29C18 and MC145503 to Z89121/921

4

CODEC INTERFACE (Continued)

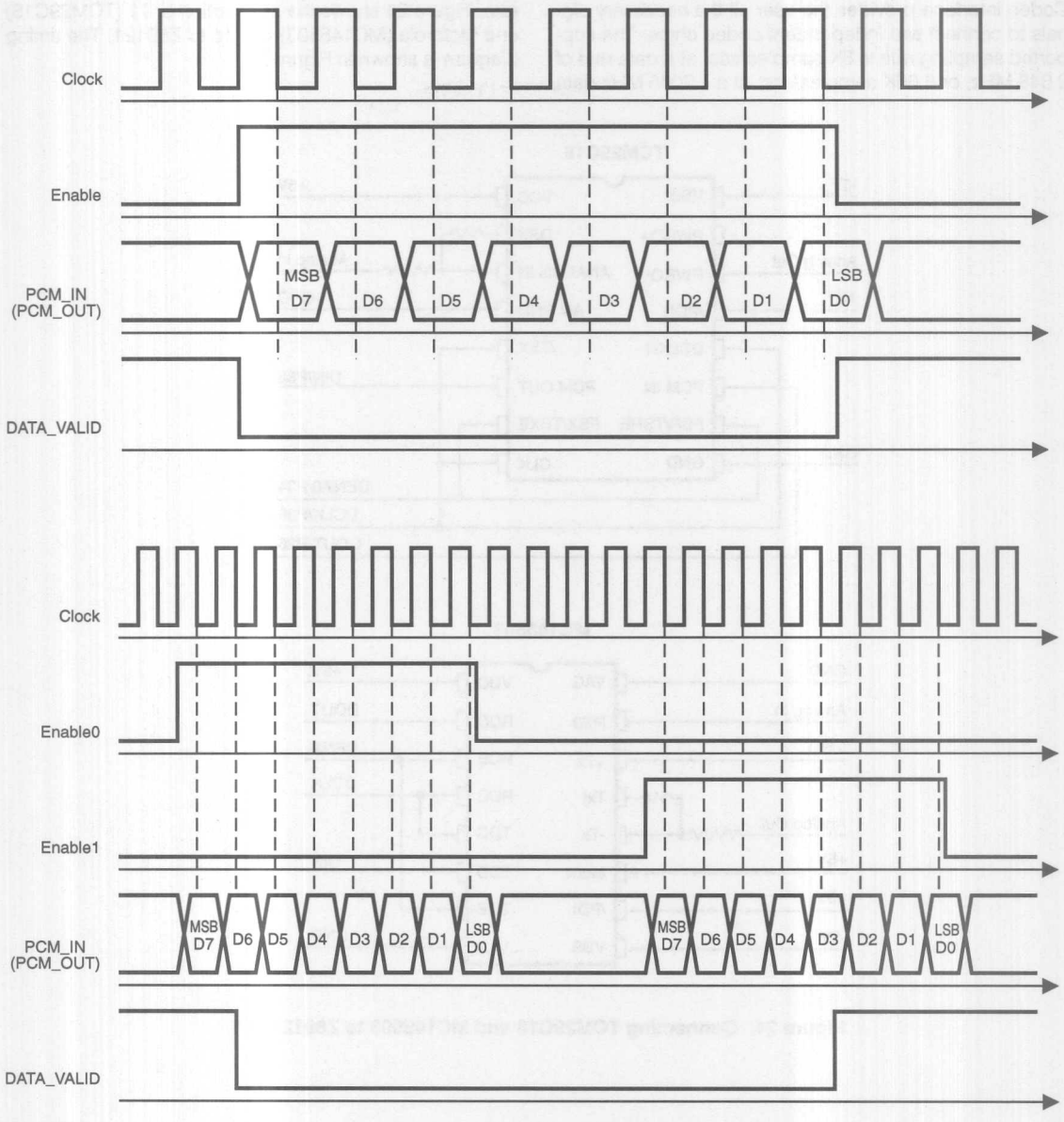


Figure 25. Timing Diagram of Codec Interface

D/A (PWM) Converter/Codec interface Register -EXT5

External DSP register EXT5 is used by the D/A converter and an External Codec Interface. The accessibility of all these devices is driven by the Analog Control register (EXT6).

The D/A converter (10-bit PWM) will be loaded by writing to register EXT5 of the DSP.

Two different Codecs can be addressed by the Analog Control register (EXT6). The data loaded to Codec0 and Codec1 is defined by writing to the EXT5 register of the DSP, while reading from this register gives the data received from Codecs.

Because the same logical register (EXT5) can be either the source or the destination for several physical devices (D/A and Codecs), the user must specify which one of all available devices he desires to write (read) to (from). EXT5 bits 'e' and 'f' are used to distinguish between different devices upon writing data to D/A, Codec0 and Codec1, as shown below. Upon reading from EXT5, the DSP reads in sequence all active (enabled) devices according to the definition of the Select_Sequence field (bits 'c' and 'd') in ACR (EXT6). The sequence of reading data can be reset by writing a 1 to the Reset_Toggle field of EXT6.

Register EXT5 is accessible to the DSP only.

Digital to Analog Converter - EXT5 (when written)

The D/A conversion is DSP driven by sending 10-bit data to the external register EXT5 of the DSP. The six remaining bits of EXT5 are reserved, as shown in the following table.

Data will be loaded into the D/A latch during the clock cycle following the (ld EXT5,data) instruction.

Table 18. EXT5 (when written)

Field	Position	Attrib.	Value	Label
Data	f----------	W	0	Should be '0'
	-edcba----------			Reserved
	------98765-----	W	%NN	DataToPWM (High Val)
	-----------43210	W	%NN	DataToPWM (Low Val)

Codec Interface Controller - EXT5 (when written)

The two Data registers of the External Codec interface are mapped into the external register EXT5 of the DSP. The eight remaining bits of EXT5 are reserved as shown in the Table 19. Data will be loaded into the corresponding Data register (defined by field 'e') during the clock cycle following the (ld EXT5,data) instruction.

Table 19. EXT5 (when written)

Field	Position	Attrib.	Value	Label
Data	f----------		1	Should be '1'
	-e----------		0	Codec0
			1	Codec1
	--dcba98--------			Reserved
	--------76543210		%NN	DataToCodec

Codec Interface Controller - EXT5 (when read)

8-Bit Data can be read from the Codec by the DSP via the external register, EXT5. Of the 16 bits of the EXT5, only eight bits, 0 through 7, return Data; the remaining bits are padded with zeroes.

Table 20. EXT5 (when read)

Field	Position	Attrib.	Value	Label
Data	fedcba98--------			Return '0'
	--------76543210		%NN	DataFromCodec

Analog Control Register (ACR)

The Analog Control register is mapped to register EXT6 of the DSP (Table 21). This read/write register is accessible by the DSP only.

The 16-bit field of EXT6 defines modes of both the A/D and the D/A. The High Byte configures the Codec.

Table 21. EXT6 Analog Control Register (ACR)

Field	Position	Attrib.	Value	Label
MPX_DSP_INT0	f---------------	R/W	1	P26
			0	Timer3
Reset_Toggle	-e--------------	R		Return '0'
	-e--------------	W	1	Reset Toggle
			0	No Effect
Select_Sequence	--dc------------	R/W	XX	Selects Codec0/Codec1 upon Reading EXT5
Reserved	----b-----------	R		Return '0'
		W		No Effect
D/A_SamplingRate	-----a98--------	R/W	11x	Reserved
			101	Reserved
			100	64 kHz
			010	16 kHz
			011	10 kHz
			001	4 kHz
			000	Reserved
Div10/12	--------7-------	R/W	1	Divided by 10
			0	Divided by 12
Reserved	---------6------	R/W		Should Be Set to '0'
Reserved	----------543210	R	%DD	Return '0'
		W		No Effect

DSP IRQ0. Defines the source of DSP IRQ0 interrupt.

Select_Sequence. Defines the Codec0 and Codec1 enabling/disabling and the sequence of reading data from these devices starting from the reset condition (Table 22).

A 1 should be written to bit 'e' in order to reset the sequence. Writing 1 to bit e ensures the next data read from EXT5 is the data of Codec0.

Table 22. Select_Sequence

Select Sequence d	Select Sequence c	Codec Enabled/Disabled Codec0	Codec Enabled/Disabled Codec1	Sequence of Access First	Sequence of Access Second
0	0	Disable	Disable	N/A	
0	1	Enable	Disable	Codec0	N/A
1	0	Enable	Enable	Codec0	Codec1
1	1	Disable	Disable	Reserved	Reserved

Div 10/12. This bit defines the speed of the Codecs. If the bit is set to 1, the Codec clock frequency is set to 2.048 MHz, and the sampling rate is 8 kHz. If the bit is reset to 0, Codec clock frequency is set to 1.7066 MHz and the sampling rate to 6.66 kHz.

Note: Bit 6 of ACR should be set to zero.

D/A_Sampling Rate. This field defines the sampling rate of the D/A output. It changes the period to Timer3 interrupt and the maximum possible accuracy of the D/A (Table 23).

Table 23. D/A Data Accuracy

D/A_Sampling Rate			Sampling Rate	D/A Accuracy
1	0	0	64 kHz	8 Bits
0	1	0	16 kHz	10 Bits
0	1	1	10 kHz	10 Bits
0	0	1	4 kHz	10 Bits

DSP Timers

Timer2 is a free-running counter that divides the XTAL frequency (20.48 MHz) to support different sampling rates for the A/D converter. The sampling rate is defined by the Analog Control Register. Upon reaching the end of a count, the timer generates an interrupt request to the DSP.

Analogous to Timer2, Timer3 generates the different sampling rates for the D/A converter. Timer3 also generates an interrupt request to the DSP upon reaching its final count value (Figure 26).

Note: The crystal speed in this example is 20.48 MHz, which is the maximum tested speed, but other lower speeds may be used.

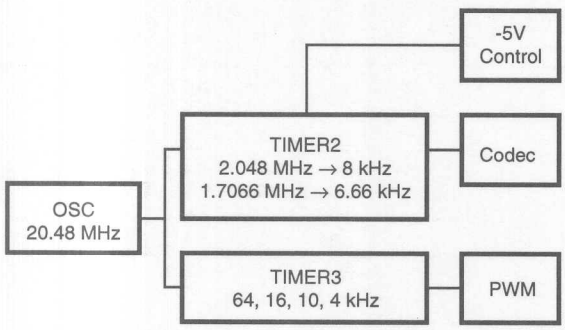


Figure 26. Timer2 and Timer3

Minus 5V DC Generation

Some Codecs require a ±5V power supply. The Z89121/921 provides a $-5V_{OUT}$ output to drive an external pump circuit. A complete circuit diagram for the –5V generation is shown in Figure 27. The reference voltage of 2.5V is generated by a resistor divider R5, R6 on the P33 input of Z86C67/921. This voltage is compared with the voltage of the voltage divider formed by R2, R4. If the latter voltage rises above the reference voltage, the comparator (inside Z86C67/921) will be switched and connect the internal 128 kHz output of Timer2 to the $-5V_{OUT}$ output pin of Z89121/921. On the contrary, the $-5V_{OUT}$ will be switched off if the voltage from voltage divider R2, R4 drops below the reference voltage. This regulates the voltage across C1 to be –5V.

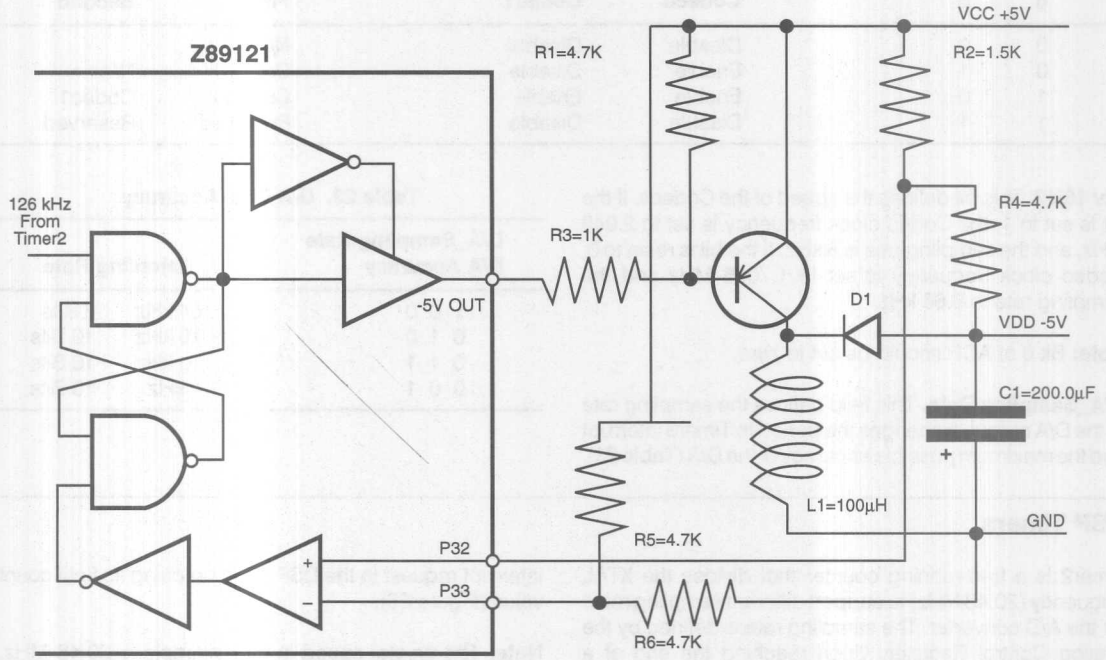


Figure 27. Circuit Diagram for -5V Generation

DRAM INTERFACE

The DRAM interface controller accepts a wide variety of external DRAM configurations (up to 48 Mbits) with 4-bit wide data buses. It can be reconfigured from the software to support: 1 Mbit x 1, 4 Mbit x 1, 1 Mbit x 4, 4 Mbit x 4 DRAM. DRAM interface registers are mapped to expanded register file (bank0A).

Table 24. Registers of DRAM Interface

Field	Position	Attrib	Value	Label
Data (Register (A)00)	76543210	R/W	%FF	Data
Control (Register (A)01)	76543210	R/W	%FF	See Text
Most Significant Byte (Register (A)02)	76543210	R/W	%FF	Data
Middle Significant Byte (Register (A)03)	76543210	R/W	%FF	Data
Least Significant Byte (Register (A)04)	76543210	R/W	%FF	Data
Refresh Count (Register (A)05)	76543210	R/W	%FF	Data

Data Register. This register is used as a logical device for reading (writing) data from (to) the DRAM. After reading by the Z8 in Auto Increment mode, the logical DRAM address specified by register (AH)04H is increased by 1 and new DRAM data at this address will be read and stored into the data register. When data is written to this register, it will be stored into the last valid DRAM logical address. The hardware write-data-to-DRAM cycle is implemented as an early write cycle with Twcs > 40 ns. The user has to load a 23-bit address into the Least, Middle, and Most Significant Byte Registers and then write the 8-bit data to the Data Register. The data will be automatically separated into higher nibble and lower nibble and stored into two subsequent locations in the DRAM (2*Address for higher nibble and 2*Address+1 for lower nibble). Writing data to the Data Register with the Auto-incremental Bit (bit 0) of the DRAM Control Register equal to 0 increases the address in the Least Significant DRAM register (AH)04H by 1.

Most, Middle, and Least Significant Byte Registers. The 23-bit logical address of DRAM is stored in these three registers. Upon writing to these registers, the read cycle from DRAM is executed so that the new data is available in the data register.

Refresh Count Register. The /RAS-only refresh cycle is transparent to user and is supported by hardware logic. This register specifies how many rows of memory matrix, starting from the beginning of the DRAM (logical address 000000H), should be refreshed. The number of the rows in DRAM to be refreshed is defined by the value in Refresh Count Register plus one and then multiplied by eight.

4

The basic timing diagram of the DRAM interface is shown in Figure 28.

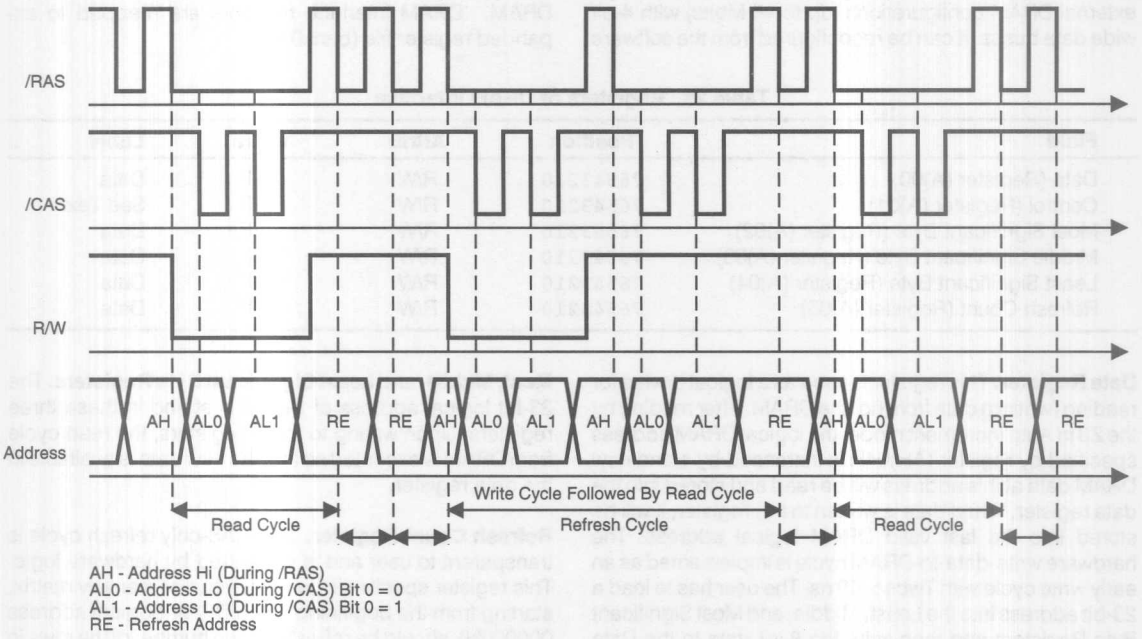


Figure 28. Timing Diagram for DRAM Interface

DRAM Control Register

The register defines DRAM access time, DRAM memory size, refresh operation, etc. (Figure 29). After Power-On Reset, the DRAM Control Register is set to %00, which defines 1 Mbit DRAM configuration with permanently active DRAM refreshing.

Table 25. DRAM Control Register

Register	Position	Attrib.	Value	Description
Access_Time	7-------	R/W	0	400 ns
			1	200 ns
ARAM_size	-6------	R/W	0	1 Mbit
			1	4 Mbit
Reserved	--54----	R/W	%DD	number These two bits can be used as User defined flags.
Refresh_start	----32--	R/W	00	Permanently
			01	Upon T0
			10	Upon T0
			11	Refresh off
Refresh_clear	------1-	R		Return '0'
		W	1	Refresh clear
			0	No effect
Autoincrement	-------0	R/W	0	Increment ON
			1	Increment OFF

Access_time. This bit defines the speed of DRAM Controller. The read/write cycle width can be changed to support slower DRAMs. When set to 1, the width of /CAS signal is set to 200 ns. Reset the Access_time bit to 0 set the width of /CAS signal to 400 ns.

DRAM_size. DRAM interface supports four different sizes of ARAM: 1 Mbit x 1, 1 Mbit x 4, 4 Mbit x 1 and 4 Mbit x 4. These require either 11- or 10-bit address bus. For 1 Mbit x 1 or 1 Mbit x 4 DRAM, the ADDR10 is used to generate select (/CAS) signal.

Bit 6	/CAS	ARAM_SEL1	ARAM_SEL0	Addr10
0	1st /CAS	3rd /CAS	2nd /CAS	Addr10
1	1st /CAS	3rd /CAS	2nd /CAS	4th /CAS

Auto Increment. This bit specifies the Auto Increment of the LBS byte of the DRAM address. The Auto Increment function does not affect any flag of Z8.

DRAM Interface

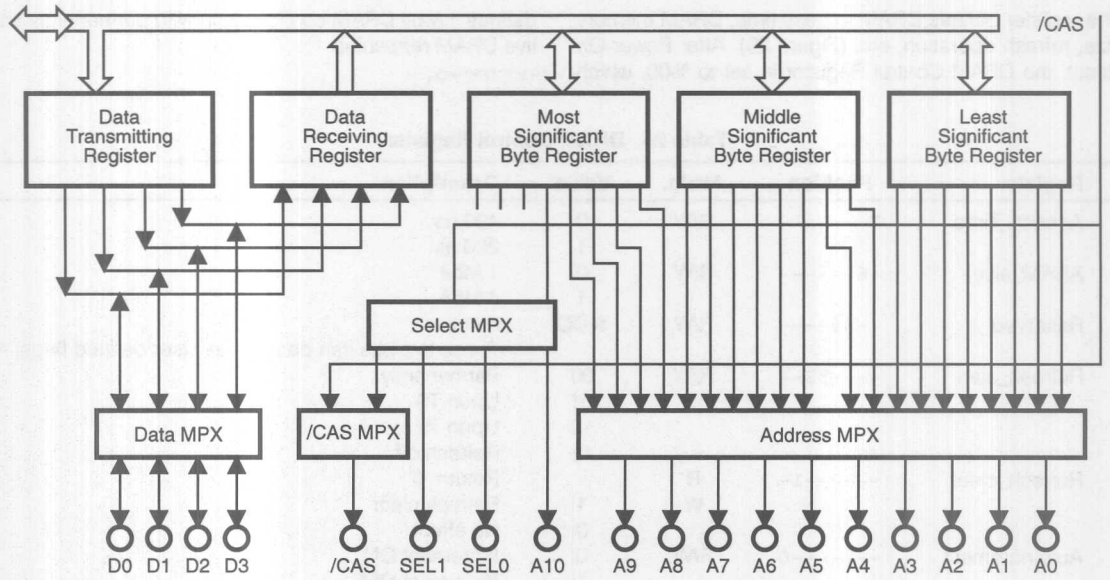


Figure 29. Block Diagram of the DRAM Interface

ABSOLUTE MAXIMUM RATINGS

Symbol	Description	Min	Max	Units
V_{CC}	Supply Voltage (*)	-0.3	+7.0	V
T_{STG}	Storage Temp	-65°	+150°	C
T_A	Oper Ambient Temp		†	C

Notes:
* Voltage on all pins with respect to GND.
† See Ordering Information.

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for an extended period may affect device reliability.

STANDARD TEST CONDITIONS

The characteristics listed below apply for standard test conditions as noted. All voltages are referenced to GND. Positive current flows into the referenced pin (Figure 30).

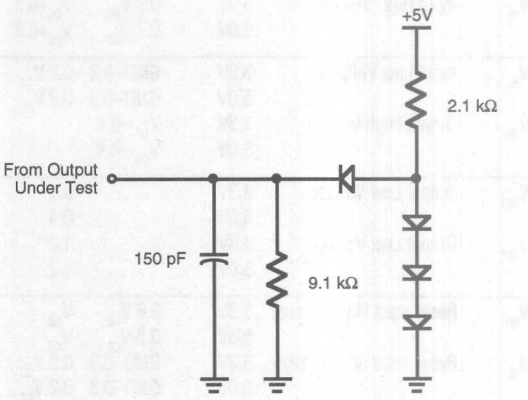


Figure 30. Test Load Diagram

CAPACITANCE

T_A = 25°C, V_{CC} = GND = 0V, f = 1.0 MHz, unmeasured pins returned to GND.

Parameter	Max
Input capacitance	12 pF
Output capacitance	12 pF
I/O capacitance	12 pF

DC ELECTRICAL CHARACTERISTICS

Symbol	Parameter	V_{CC} Note [1]	T_A = 0°C to +70°C Min	Max	Typical @ 25°C	Units
I_{CC}	Supply Current	5.0V		65	40	mA
I_{CC1}	HALT Mode Current	5.0V		10	6	mA
I_{CC2}	STOP Mode Current	5.0V		20	6	µA
I_{-5V}	Output Current, –5V Supply		–15		–20	mA

Note:
[1] 5.0V ±0.5V

DC ELECTRICAL CHARACTERISTICS

Sym	Parameter	V_{CC} Note [3]	T_A = 0°C to +70°C Min	Max	T_A = –40°C to +105°C Min	Max	Typical @ 25°C	Units	Conditions
	Max Input Voltage	3.3V		7		7		V	I_{IN} 250 uA
		5.0V		7		7		V	I_{IN} 250 uA
V_{CH}	Clock Input High Voltage	3.3V	0.7 V_{CC}	V_{CC}+0.3	0.7 V_{CC}	V_{CC}+0.3	1.3	V	Driven by External Clock Generator
		5.0V	0.7 V_{CC}	V_{CC}+0.3	0.7 V_{CC}	V_{CC}+0.3	2.5	V	Driven by External Clock Generator
V_{CL}	Clock Input Low Voltage	3.3V	GND–0.3	0.2 V_{CC}	GND–0.3	0.2 V_{CC}	0.7	V	Driven by External Clock Generator
		5.0V	GND–0.3	0.2 V_{CC}	GND–0.3	0.2 V_{CC}	1.5	V	Driven by External Clock Generator
V_{IH}	Input High Voltage	3.3V	0.7 V_{CC}	V_{CC}+0.3	0.7 V_{CC}	V_{CC}+0.3	1.3	V	
		5.0V	0.7 V_{CC}	V_{CC}+0.3	0.7 V_{CC}	V_{CC}+0.3	2.5	V	
V_{IL}	Input Low Voltage	3.3V	GND–0.3	0.2 V_{CC}	GND–0.3	0.2 V_{CC}	0.7	V	
		5.0V	GND–0.3	0.2 V_{CC}	GND–0.3	0.2 V_{CC}	1.5	V	
V_{OH}	Output High Voltage	3.3V	V_{CC} –0.4		V_{CC} –0.4		3.1	V	I_{OH} = –2.0 mA
		5.0V	V_{CC} –0.4		V_{CC} –0.4		4.8	V	I_{OH} = –2.0 mA
V_{OL1}	Output Low Voltage	3.3V		0.6		0.6	0.2	V	I_{OL} = +4.0 mA
		5.0V		0.4		0.4	0.1	V	I_{OL} = +4.0 mA
V_{OL2}	Output Low Voltage	3.3V		1.2		1.2	0.3	V	I_{OL} = +6 mA, 3 Pin Max
		5.0V		1.2		1.2	0.3	V	I_{OL} = +12 mA, 3 Pin Max
V_{RH}	Reset Input High Voltage	3.3V	0.8 V_{CC}	V_{CC}	0.8 V_{CC}	V_{CC}	1.5	V	
		5.0V	0.8 V_{CC}	V_{CC}	0.8 V_{CC}	V_{CC}	2.1	V	
V_{RI}	Reset Input Low Voltage	3.3V	GND–0.3	0.2 V_{CC}	GND–0.3	0.2 V_{CC}	1.1		
		5.0V	GND–0.3	0.2 V_{CC}	GND–0.3	0.2 V_{CC}	1.7		
V_{OFFSET}	Comparator Input Offset	3.3V		25		25	10	mV	
	Voltage	5.0V		25		25	10	mV	
I_{IL}	Input Leakage	3.3V	–1	1	–1	2	<1	µA	V_{IN} = 0V, V_{CC}
		5.0V	–1	1	–1	2	<1	µA	V_{IN} = 0V, V_{CC}
I_{OL}	Output Leakage	3.3V	–1	1	–1	2	<1	µA	V_{IN} = 0V, V_{CC}
		5.0V	–1	1	–1	2	<1	µA	V_{IN} = 0V, V_{CC}
I_{IR}	Reset Input Current	3.3V		–45		–60	–20	µA	
		5.0V		–55		–70	–30	µA	

AC CHARACTERISTICS

External I/O or Memory Read and Write Timing Diagram

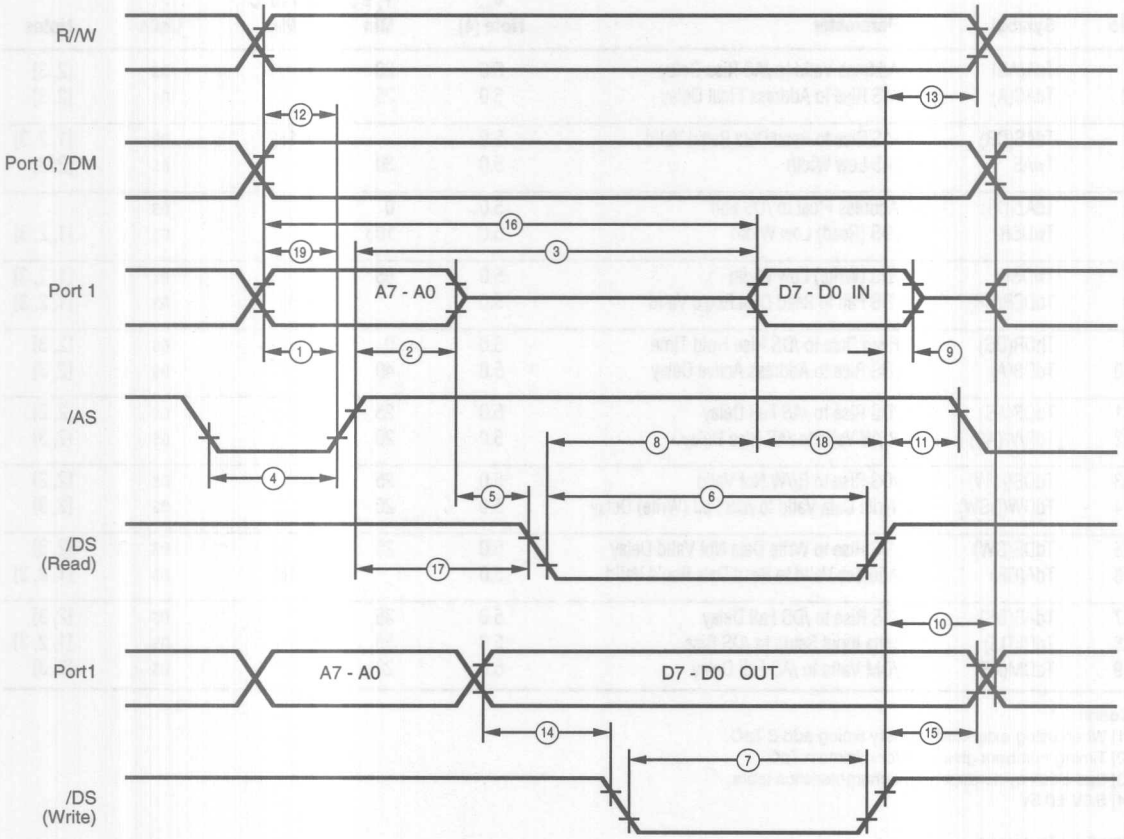


Figure 31. External I/O or Memory Read/Write Timing

AC CHARACTERISTICS

External I/O or Memory Read and Write Timing Table

No	Symbol	Parameter	V_{CC} Note [4]	T_A = 0°C to +70°C Min	Max	Units	Notes
1	TdA(AS)	Address Valid to /AS Rise Delay	5.0	20		ns	[2, 3]
2	TdAS(A)	/AS Rise to Address Float Delay	5.0	25		ns	[2, 3]
3	TdAS(DR)	/AS Rise to Read Data Req'd Valid	5.0		150	ns	[1, 2, 3]
4	TwAS	/AS Low Width	5.0	30		ns	[2, 3]
5	TdAZ(DS)	Address Float to /DS Fall	5.0	0		ns	
6	TwDSR	/DS (Read) Low Width	5.0	105		ns	[1, 2, 3]
7	TwDSW	/DS (Write) Low Width	5.0	65		ns	[1, 2, 3]
8	TdDSR(DR)	/DS Fall to Read Data Req'd Valid	5.0		55	ns	[1, 2, 3]
9	ThDR(DS)	Read Data to /DS Rise Hold Time	5.0	0		ns	[2, 3]
10	TdDS(A)	/DS Rise to Address Active Delay	5.0	40		ns	[2, 3]
11	TdDS(AS)	/DS Rise to /AS Fall Delay	5.0	25		ns	[2, 3]
12	TdR/W(AS)	R//W Valid to /AS Rise Delay	5.0	20		ns	[2, 3]
13	TdDS(R/W)	/DS Rise to R//W Not Valid	5.0	25		ns	[2, 3]
14	TdDW(DSW)	Write Data Valid to /DS Fall (Write) Delay	5.0	20		ns	[2, 3]
15	TdDS(DW)	/DS Rise to Write Data Not Valid Delay	5.0	25		ns	[2, 3]
16	TdA(DR)	Address Valid to Read Data Req'd Valid	5.0		180	ns	[1, 2, 3]
17	TdAS(DS)	/AS Rise to /DS Fall Delay	5.0	35		ns	[2, 3]
18	TdDI(DS)	Data Input Setup to /DS Rise	5.0	50		ns	[1, 2, 3]
19	TdDM(AS)	/DM Valid to /AS Fall Delay	5.0	20		ns	[2, 3]

Notes:
[1] When using extended memory timing add 2 TpC.
[2] Timing numbers given are for minimum TpC.
[3] See clock cycle dependent characteristics table.
[4] 5.0V ±0.5V

Standard Test Load
All timing references use 0.9 V_{CC} for a logic1 and 0.1 V_{CC} for a logic0.

AC ELECTRICAL CHARACTERISTICS
Additional Timing Diagram

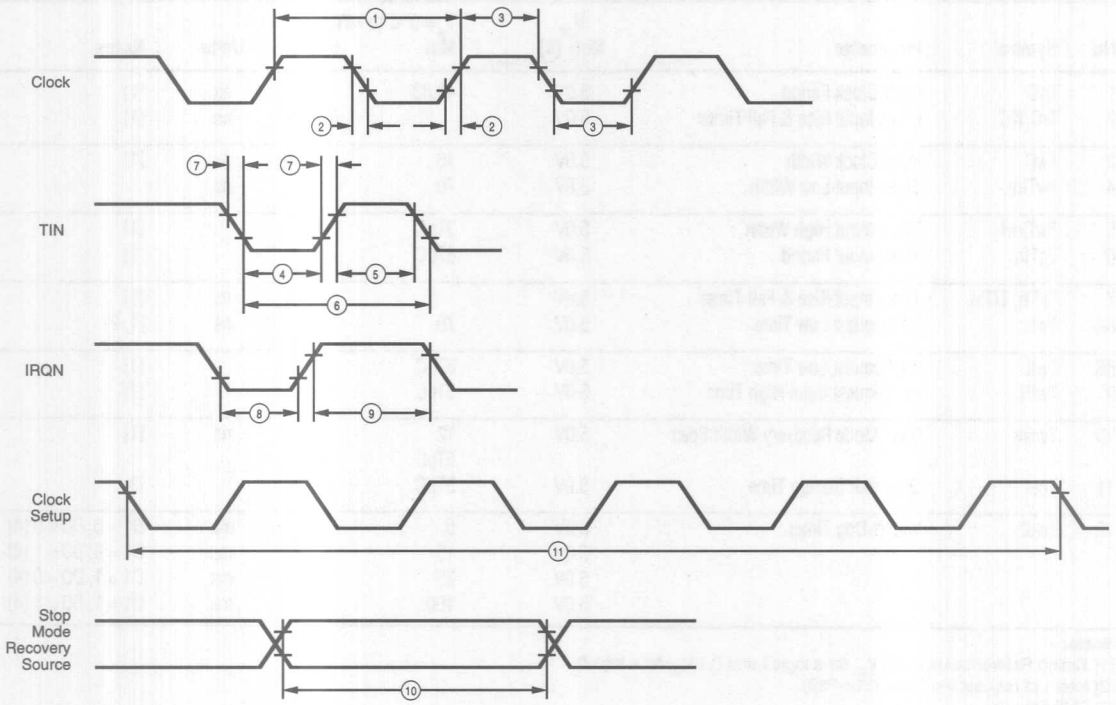


Figure 32. Additional Timing

AC ELECTRICAL CHARACTERISTICS
Additional Timing Table

No	Symbol	Parameter	V_{CC} Note [5]	T_A = 0°C to +70°C Min	Max	Units	Notes
1	TpC	Input Clock Period	5.0V	48.83		ns	[1]
2	TrC,TfC	Clock Input Rise & Fall Times	5.0V		6	ns	[1]
3	TwC	Input Clock Width	5.0V	16		ns	[1]
4	TwTinL	Timer Input Low Width	5.0V	70		ns	
5	TwTinH	Timer Input High Width	5.0V	3TpC			[1]
6	TpTin	Timer Input Period	5.0V	8TpC			[1]
7	TrTin, TfTin	Timer Input Rise & Fall Timer	5.0V		100	ns	[1]
8A	TwIL	Int. Request Low Time	5.0V	70		ns	[1, 2]
8B	TwIL	Int. Request Low Time	5.0V	3TpC			[1]
9	TwIH	Int. Request Input High Time	5.0V	3TpC			[1]
10	Twsm	Stop-Mode Recovery Width Spec	5.0V	12 5TpC		ns	[1]
11	Tost	Oscillator Startup Time	5.0V	5TpC			[3]
12	Twdt	Watch-Dog Timer	5.0V	5		ms	D1 = 0, D0 = 0 [4]
			5.0V	15		ms	D1 = 0, D0 = 1 [4]
			5.0V	25		ms	D1 = 1, D0 = 0 [4]
			5.0V	100		ms	D1 = 1, D0 = 1 [4]

Notes:
[1] Timing Reference uses 0.9 V_{CC} for a logic1 and 0.1 V_{CC} for a logic0.
[2] Interrupt request via Port 3 (P31-P33).
[3] SMR-D5 = 0.
[4] Reg. WDT.
[5] 5.0V ±0.5V

AC ELECTRICAL CHARACTERISTICS
Handshake Timing Diagrams

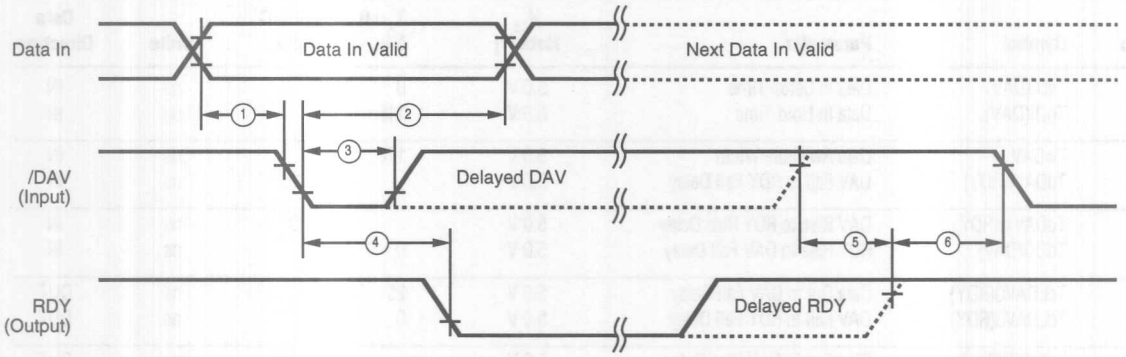


Figure 33. Input Handshake Timing

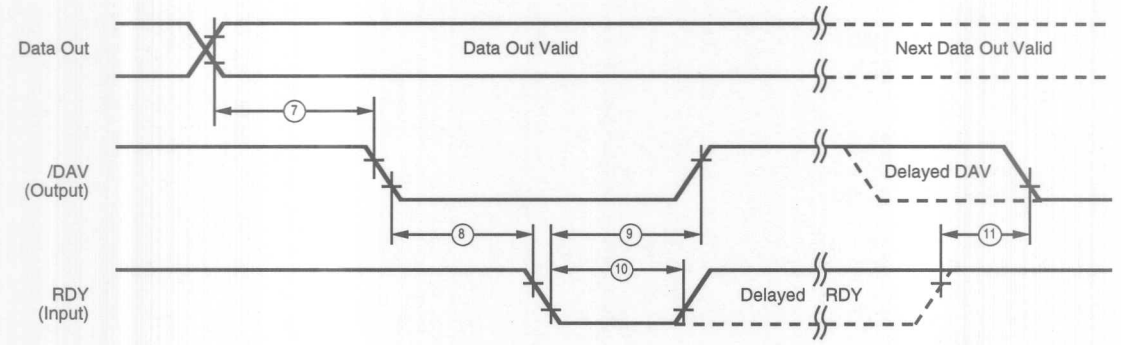


Figure 34. Output Handshake Timing

4

AC ELECTRICAL CHARACTERISTICS

Handshake Timing Table

No	Symbol	Parameter	V_{CC} Note [1]	T_A = 0°C to +70°C Min	Max	Units	Data Direction
1	TsDI(DAV)	Data In Setup Time	5.0 V	0		ns	IN
2	ThDI(DAV)	Data In Hold Time	5.0 V	115		ns	IN
3	TwDAV	Data Available Width	5.0 V	110		ns	IN
4	TdDAVI(RDY)	DAV Fall to RDY Fall Delay	5.0 V		115	ns	IN
5	TdDAVId(RDY)	DAV Rise to RDY Rise Delay	5.0 V		80	ns	IN
6	TdDO(DAV)	RDY Rise to DAV Fall Delay	5.0 V	0		ns	IN
7	TcLDAV0(RDY)	Data Out to DAV Fall Delay	5.0 V	25		ns	OUT
8	TcLDAV0(RDY)	DAV Fall to RDY Fall Delay	5.0 V	0		ns	OUT
9	TdRDY0(DAV)	RDY Fall to DAV Rise Delay	5.0 V		115	ns	OUT
10	TwRDY	RDY Width	5.0 V	80		ns	OUT
11	TdRDY0d(DAV)	RDY Rise to DAV Fall Delay	5.0 V		80	ns	OUT

Note:
[1] 5.0V ±0.5V

Z8 EXPANDED REGISTER FILE REGISTERS

Expanded Register Bank B

Register	Position	Attrib.	Value	Description
Outgoing Reg. to DSP EXT0 (High Byte) (B)%00	76543210	R/W		DSP EXT0, Bits D15-D8
Outgoing Reg. to DSP EXT0 (Low Byte) (B)%01	76543210	R/W		DSP EXT0, Bits D7-D0
Outgoing Reg. to DSP EXT1 (High Byte) (B)%02	76543210	R/W		DSP EXT1, Bits D15-D8
Outgoing Reg. to DSP EXT1 (Low Byte) (B)%03	76543210	R/W		DSP EXT1, Bits D7-D0
Outgoing Reg. to DSP EXT2 (High Byte) (B)%04	76543210	R/W		DSP EXT2, Bits D15-D8
Outgoing Reg. to DSP EXT2 (Low Byte) (B)%05	76543210	R/W		DSP EXT2, Bits D7-D0
Outgoing Reg. to DSP EXT3 (High Byte) (B)%06	76543210	R/W		DSP EXT3, Bits D15-D8
Outgoing Reg. to DSP EXT3 (Low Byte) (B)%07	76543210	R/W		DSP EXT3, Bits D7-D0

Z8 EXPANDED REGISTER FILE REGISTERS

Expanded Register Bank B (Continued)

Register	Position	Attrib.	Value	Description
Incoming Reg. to DSP EXT0 (High Byte) (B)%08	76543210	R		DSP EXT0, Bits D15-D8
Incoming Reg. to DSP EXT0 (Low Byte) (B)%09	76543210	R		DSP EXT0, Bits D7-D0
Incoming Reg. to DSP EXT1 (High Byte) (B)%0A	76543210	R		DSP EXT1, Bits D15-D8
Incoming Reg. to DSP EXT1 (Low Byte) (B)%0B	76543210	R		DSP EXT1, Bits D7-D0
Incoming Reg. to DSP EXT2 (High Byte) (B)%0C	76543210	R		DSP EXT2, Bits D15-D8
Incoming Reg. to DSP EXT2 (Low Byte) (B)%0D	76543210	R		DSP EXT2, Bits D7-D0
Incoming Reg. to DSP EXT3 (High Byte) (B)%0E	76543210	R		DSP EXT3, Bits D15-D8
Incoming Reg. to DSP EXT3 (Low Byte) (B)%0F	76543210	R		DSP EXT3, Bits D7-D0

Expanded Register Bank F

Register	Position	Attrib.	Value	Description
PCON (F)%00				
	76543---			Reserved
	-----2--	R	0	Port 1 Open-drain
			1	Port 1 Push-pull Active*
	------1-	R	0	Port 0 Open-drain
			1	Port 0 Push-pull Active*
	-------0	R	0	P34, P35 Standard Output*
			1	P34, P35 Comparator Output
DSPCON (F)0CH				
Z8_SCLK	76------	R/W	00	2.5 MHz (OSC/8)
			01	5 MHz (OSC/4)
			1x	10 MHZ (OSC/2)
DSP_Reset	--5-----	R		Return '0'
		W	0	No effect
			1	Reset DSP
DSP_Run	---4----	R/W	0	Halt_DSP
			1	Run_DSP
Reserved	----32--		xx	
				Return '0'
				No effect
IntFeedback	------1-	R		FB_DSP_INT2
		W	1	Set DSP_INT2
			0	No effect
	-------0	R		FB_Z8_IRQ3
		W	1	Clear IRQ3
			0	No effect
P4 (F)%02	76543210	R/W	%NN	Port 4 Data Register
P4M (F)%03	76543210	R	%FF	Returns %FF
		W	0	Defines P4X as Output
			1	Defines P4X as Input
P5 (F)%04	76543210	R/W	%NN	Port 5 Data Register
P5M (F)%05	76543210	R	%FF	Returns %FF
		W	0	Defines P5X pin as Output
			1	Defines P5X pin as Input
P45CON (F)%06				
	765-321-			Reserved
	---4----	W	0	Port 5 Open-drain
			1	Port 5 Push-pull Active*
	-------0	W	0	Port 4 Open-drain
			1	Port 4 Push-pull Active*

Note:
* Default setting after Reset

4

Z8 EXPANDED REGISTER FILE REGISTERS

Expanded Register Bank F (Continued)

Register	Position	Attrib.	Value	Description
SMR (F)%0B				
	7-------	R	0	POR*
			1	Stop Recovery
	-6------	W	0	Low Stop Recovery Level*
			1	High Stop Recovery Level
	--5-----	W	0	Stop Delay On*
			1	Stop Delay Off
	---432--	W		Stop-Mode Recovery Source
			000	POR Only*
			001	Reserved
			010	P31
			011	P32
			100	P33
			101	P27
			110	P2 NOR 0-3
			111	P2 NOR 0-7
	------1-			Reserved
	-------0	W	0	SCLK/TCLK Not Divide by 16†
			1	SCLK/TCLK Divide by 16
WDTMR (F)%0F				
	765-----			Reserved
	---4----	R/W	0	On-Board RC for WDT*
			1	XTAL for WDT
	----3---	R/W	0	WDT Off During STOP
			1	WDT On During STOP*
	-----2--	R/W	0	WDT Off During HALT
			1	WDT On During HALT*
	------10	R/W		Int RC Osc Ext Clock
			00	5 ms 256 TpC
			01*	15 ms 512 TpC
			10	25 ms 1024 TpC
			11	100 ms 4096 TpC

Notes:
* Default setting after Reset
† Reset after Stop Mode Recovery

Z8 CONTROL REGISTERS

Register	Position	Attrib.	Value	Description
%F0	76543210			Reserved
TMR %F1				T_{OUT} Modes
	76------	RW	00	Not Used
			01	T0 Out
			10	T1 Out
			11	Internal Clock Out P36
				T_{IN} Modes
	--54----	RW	00	External Clock Input
			01	Gate Input
			10	Trigger Input (Non-Retriggerable)
			11	Trigger Input (Retriggerable)
	----3---	R/W	0	Disable T1 Count
			1	Enable T1 Count
	-----2--	R/W	0	No Effect
			1	Load T1
	------1-	R/W	0	Disable T0 Count
			1	Enable T0 Count
	-------0	R/W	0	No Effect
			1	Load T0
T1 %F2	76543210	R	%NN	T1 Current Value
		W	%NN	T1 Initial Value
PRE1 %F3	765432--	W		Prescaler Modulo (1-64 Dec)
	------1-			T1 Clock Source
		W	0	External Timing Input (T_{IN}) Mode
			1	Internal Clock
	-------0			T1 Count Mode
		W	0	Single Pass
			1	Modulo N
T0 %F4	76543210	R	%NN	T0 Current Value
		W	%NN	T0 Initial Value
PRE0 %F5	765432--	W		Prescaler Modulo (1-64 Dec)
	------1-			Reserved
	-------0			T0 Count Mode
		W	0	Single Pass
			1	Modulo N
P2M %F6	76543210	W	0	Defines P2X pin as Output
			1	Defines P2X pin as Input

4

Z8 CONTROL REGISTERS (Continued)

Register	Position	Attrib.	Value	Description
P3M %F7				
	7-------			Reserved
	-6------	W	0	P30 = Input; P37 = Output
	--5-----	W	0	P31 = Input (T_{IN}); P36 = Output (T_{OUT})*
			1	P31 = /DAV2/RDY2; P36 = RDY2//DAV2
	---43---	W	00	P33 = Input; P34 = Output*
			01	P33 = Input; P34 = /DM
			10	P33 = Input; P34 = /DM
			11	P33 = /DAV1/RDY1; P34 = RDY1//DAV1
	-----2--	W	0	P32 = Input; P35 = Output*
			1	P32 = /DAV0/RDY0; P35 = RDY0//DAV0
	------1-	W	0	P31, P32 Digital Mode
			1	P31, P32 Analog Mode
	-------0	R/W	0	Port 2 Open-drain*
			1	Port 2 Push-pull Active
P01M %F8				TOUT Modes
	76------	W		P04-P07 Mode
			00	Output
			01	Input*
			1x	A15-A12
	--5-----	W		External Memory Timing
			0	Normal*
			1	Extended
	---43---	W		P10-P17 Mode
			00	Byte Output
			01	Byte Input*
			10	AD7-AD0
			11	High-Z AD7-AD0, /AS, /DS/ R/W, A11-A8 A15-A12, If selected
	-----2--	W		Stack Selection
			0	External
			1	Internal*
	------10	W		P00-P03 Mode
			00	Output
			01	Input*
			1x	A11-A8

Note:
* Default setting after Reset.

Register	Position	Attrib.	Value	Description
IPR %F9	76------			Reserved
	--5-----	W		IRQ3, IRQ5 Priority (Group A)
			0	IRQ5 > IRQ3
			1	IRQ3 > IRQ5
	-----1--	W		IRQ0, IRQ2 Priority (Group B)
			0	IRQ2 > IRQ0
			1	IRQ0 > IRQ2
	-----2--	W		IRQ1, IRQ4 Priority (Group C)
			0	IRQ1 > IRQ4
			1	IRQ4 > IRQ1
	---43--0	W		Interrupt Group Priority
			000	Reserved
			001	C>A>B
			010	A>B>C
			011	A>C>B
			100	B>C>A
			101	C>B>A
			110	B>A>C
			111	Reserved
IRQ %FA	76------	R/W		Inter Edge (R = Rising edge; F = Falling edge)
			00	P31 = F; P32 = F
			01	P31 = F; P32 = R
			10	P31 = R; P32 = F
			11	P31 = RF; P32 = RF
	--543210	R/W		IRQ5 = T1
				IRQ4 = T0
				IRQ3 = DSP
				IRQ2 = P31 Input
				IRQ1 = P33 Input
				IRQ0 = P32 Input
IMR %FB	7-------	R/W	0	Disables Interrupts
			1	Enables Interrupts
	-6------	RW	0	Disables RAM Protect
			1	Enables RAM Protect
	--543210	R/W	0	Disables IRQ5-IRQ0 (D0 = IRQ0)
			1	Enables IRQ5-IRQ0
Flags %FC	7-------	R/W		Carry Flag
	-6------	R/W		Zero Flag
	--5-----	R/W		Sign Flag
	---4----	R/W		Overflow Flag
	----3---	R/W		Decimal Adjust Flag
	-----2--	R/W		Half Carry Flag
	------1-	R/W		User Flag F2
	-------0	R/W		User Flag F1
RP %FD	7654----	R/W	%N0	Working Register Group
	----3210	R/W	%0N	Expanded Register File Bank
SPH %FE	76543210	R/W	%NN	Stack Pointer Upper Byte
SPL %FF	76543210	R/W	%NN	Stack Pointer Lower Byte

Z8 INSTRUCTION SET NOTATION

Addressing Modes. The following notation is used to describe the addressing modes and instruction operations as shown in the instruction summary.

Symbol	Meaning
IRR	Indirect register pair or indirect working-register pair address
Irr	Indirect working-register pair only
X	Indexed address
DA	Direct address
RA	Relative address
IM	Immediate
R	Register or working-register address
r	Working-register address only
IR	Indirect-register or indirect working-register address
Ir	Indirect working-register address only
RR	Register pair or working register pair address

Symbols. The following symbols are used in describing the instruction set.

Symbol	Meaning
dst	Destination location or contents
src	Source location or contents
cc	Condition code
@	Indirect address prefix
SP	Stack Pointer
PC	Program Counter
FLAGS	Flag register (Control Register 252)
RP	Register Pointer (R253)
IMR	Interrupt mask register (R251)

Flags. Control register (R252) contains the following six flags:

Symbol	Meaning
C	Carry flag
Z	Zero flag
S	Sign flag
V	Overflow flag
D	Decimal-adjust flag
H	Half-carry flag

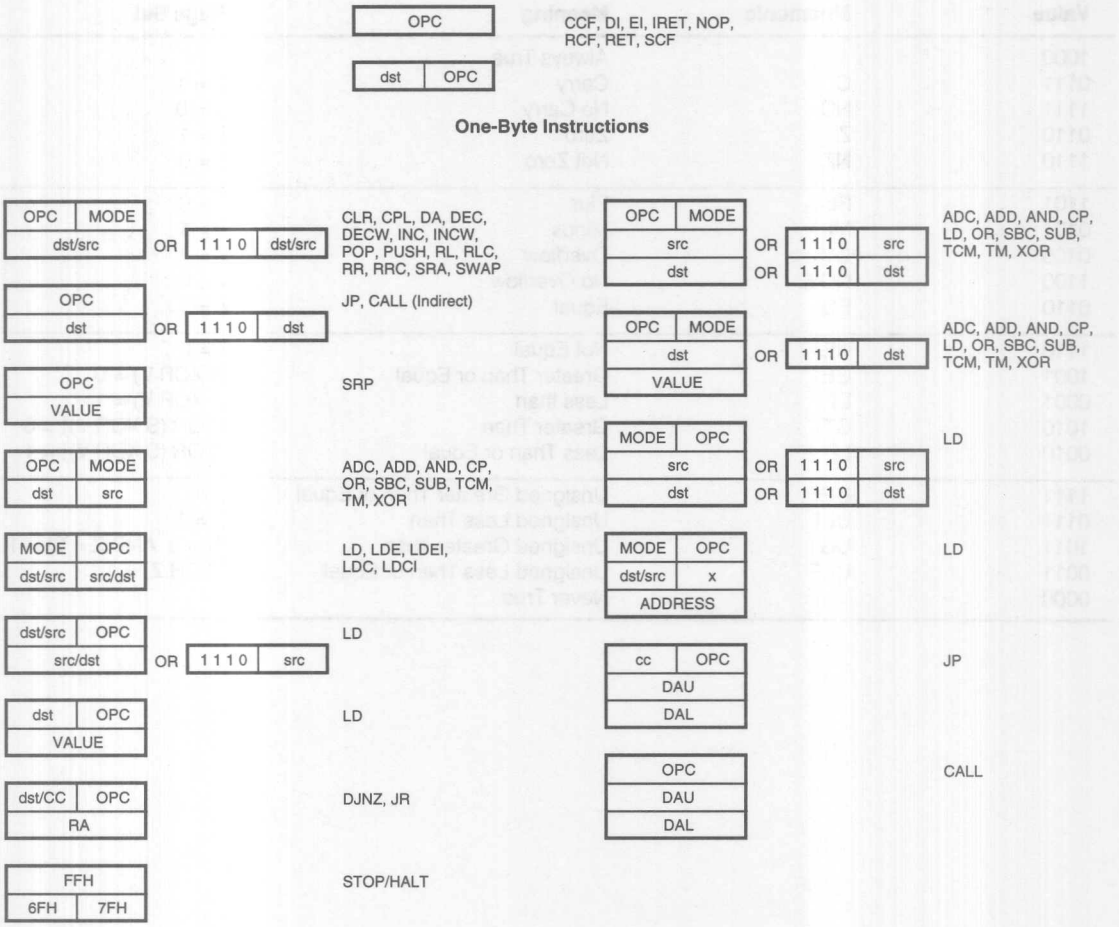
Affected flags are indicated by:

0	Clear to zero
1	Set to one
*	Set to clear according to operation
-	Unaffected
x	Undefined

CONDITION CODES

Value	Mnemonic	Meaning	Flags Set
1000		Always True	
0111	C	Carry	C = 1
1111	NC	No Carry	C = 0
0110	Z	Zero	Z = 1
1110	NZ	Not Zero	Z = 0
1101	PL	Plus	S = 0
0101	MI	Minus	S = 1
0100	OV	Overflow	V = 1
1100	NOV	No Overflow	V = 0
0110	EQ	Equal	Z = 1
1110	NE	Not Equal	Z = 0
1001	GE	Greater Than or Equal	(S XOR V) = 0
0001	LT	Less than	(S XOR V) = 1
1010	GT	Greater Than	[Z OR (S XOR V)] = 0
0010	LE	Less Than or Equal	[Z OR (S XOR V)] = 1
1111	UGE	Unsigned Greater Than or Equal	C = 0
0111	ULT	Unsigned Less Than	C = 1
1011	UGT	Unsigned Greater Than	(C = 0 AND Z = 0) = 1
0011	ULE	Unsigned Less Than or Equal	(C OR Z) = 1
0000		Never True	

INSTRUCTION FORMATS



INSTRUCTION SUMMARY

Note: Assignment of a value is indicated by the symbol " ← ". For example:

dst ← dst + src

indicates that the source data is added to the destination data and the result is stored in the destination location. The notation "addr (n)" is used to refer to bit (n) of a given operand location. For example:

dst (7)

refers to bit 7 of the destination operand.

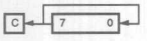
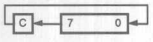
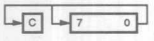
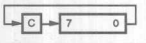
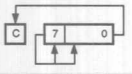
INSTRUCTION SUMMARY (Continued)

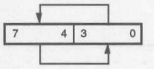
Instruction and Operation	Address Mode dst	src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
ADC dst, src dst←dst + src +C	†		1[ ]	*	*	*	*	0	*
ADD dst, src dst←dst + src	†		0[ ]	*	*	*	*	0	*
AND dst, src dst←dst AND src	†		5[ ]	-	*	*	0	-	-
CALL dst SP←SP - 2 @SP←PC, PC←dst	DA IRR		D6 D4	-	-	-	-	-	-
CCF C←NOT C			EF	*	-	-	-	-	-
CLR dst dst←0	R IR		B0 B1	-	-	-	-	-	-
COM dst dst←NOT dst	R IR		60 61	-	*	*	0	-	-
CP dst, src dst - src	†		A[ ]	*	*	*	*	-	-
DA dst dst←DA dst	R IR		40 41	*	*	*	X	-	-
DEC dst dst←dst - 1	R IR		00 01	-	*	*	*	-	-
DECW dst dst←dst - 1	RR IR		80 81	-	*	*	*	-	-
DI IMR(7)←0			8F	-	-	-	-	-	-
DJNZr, dst r←r - 1 if r ≠ 0 PC←PC + dst Range: +127, -128	RA		rA r = 0 - F	-	-	-	-	-	-
EI IMR(7)←1			9F	-	-	-	-	-	-
HALT			7F	-	-	-	-	-	-

Instruction and Operation	Address Mode dst	src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
INC dst dst←dst + 1	r R IR		rE r = 0 - F 20 21	-	*	*	*	-	-
INCW dst dst←dst + 1	RR IR		A0 A1	-	*	*	*	-	-
IRET FLAGS←@SP; SP←SP + 1 PC←@SP; SP←SP + 2; IMR(7)←1			BF	*	*	*	*	*	*
JP cc, dst if cc is true PC←dst	DA IRR		cD c = 0 - F 30	-	-	-	-	-	-
JR cc, dst if cc is true, PC←PC + dst Range: +127, -128	RA		cB c = 0 - F	-	-	-	-	-	-
LD dst, src dst←src	r r R r X r Ir R R R IR IR	Im R r X r Ir r R IR IM IM R	rC r8 r9 r = 0 - F C7 D7 E3 F3 E4 E5 E6 E7 F5	-	-	-	-	-	-
LDC dst, src	r	Irr	C2	-	-	-	-	-	-
LDCI dst, src dst←src r←r +1; rr←rr + 1	Ir	Irr	C3	-	-	-	-	-	-

4

INSTRUCTION SUMMARY (Continued)

Instruction and Operation	Address Mode dst	Address Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
NOP			FF	-	-	-	-	-	-
OR dst, src dst←dst OR src	†		4[ ]	-	*	*	0	-	-
POP dst dst←@SP; SP←SP + 1	R IR		50 51	-	-	-	-	-	-
PUSH src SP←SP - 1; @SP←src		R IR	70 71	-	-	-	-	-	-
RCF C←0			CF	0	-	-	-	-	-
RET PC←@SP; SP←SP + 2			AF	-	-	-	-	-	-
RL dst 	R IR		90 91	*	*	*	*	-	-
RLC dst 	R IR		10 11	*	*	*	*	-	-
RR dst 	R IR		E0 E1	*	*	*	*	-	-
RRC dst 	R IR		C0 C1	*	*	*	*	-	-
SBC dst, src dst←dst←src←C	†		3[ ]	*	*	*	*	1	*
SCF C←1			DF	1	-	-	-	-	-
SRA dst 	R IR		D0 D1	*	*	*	0	-	-
SRP src RP←src		Im	31	-	-	-	-	-	-

Instruction and Operation	Address Mode dst	Address Mode src	Opcode Byte (Hex)	Flags Affected C	Z	S	V	D	H
STOP			6F	-	-	-	-	-	-
SUB dst, src dst←dst←src	†		2[ ]	*	*	*	*	1	*
SWAP dst 	R IR		F0 F1	X	*	*	X	-	-
TCM dst, src (NOT dst) AND src	†		6[ ]	-	*	*	0	-	-
TM dst, src dst AND src	†		7[ ]	-	*	*	0	-	-
XOR dst, src dst←dst XOR src	†		B[ ]	-	*	*	0	-	-

† These instructions have an identical set of addressing modes, which are encoded for brevity. The first opcode nibble is found in the instruction set table above. The second nibble is expressed symbolically by a '[ ]' in this table, and its value is found in the following table to the left of the applicable addressing mode pair.

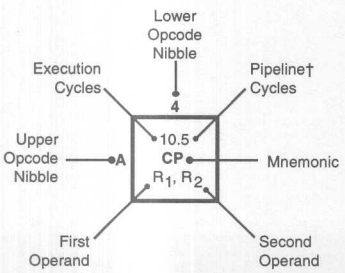
For example, the opcode of an ADC instruction using the addressing modes r (destination) and Ir (source) is 13.

Address Mode dst	Address Mode src	Lower Opcode Nibble
r	r	[2]
r	Ir	[3]
R	R	[4]
R	IR	[5]
R	IM	[6]
IR	IM	[7]

OPCODE MAP

Lower Nibble (Hex)

Upper Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	6.5 **DEC** R1	6.5 **DEC** IR1	6.5 **ADD** r1, r2	6.5 **ADD** r1, Ir2	10.5 **ADD** R2, R1	10.5 **ADD** IR2, R1	10.5 **ADD** R1, IM	10.5 **ADD** IR1, IM	6.5 **LD** r1, R2	6.5 **LD** r2, R1	12/10.5 **DJNZ** r1, RA	12/10.0 **JR** cc, RA	6.5 **LD** r1, IM	12.10.0 **JP** cc, DA	6.5 **INC** r1	
1	6.5 **RLC** R1	6.5 **RLC** IR1	6.5 **ADC** r1, r2	6.5 **ADC** r1, Ir2	10.5 **ADC** R2, R1	10.5 **ADC** IR2, R1	10.5 **ADC** R1, IM	10.5 **ADC** IR1, IM								
2	6.5 **INC** R1	6.5 **INC** IR1	6.5 **SUB** r1, r2	6.5 **SUB** r1, Ir2	10.5 **SUB** R2, R1	10.5 **SUB** IR2, R1	10.5 **SUB** R1, IM	10.5 **SUB** IR1, IM								
3	8.0 **JP** IRR1	6.1 **SRP** IM	6.5 **SBC** r1, r2	6.5 **SBC** r1, Ir2	10.5 **SBC** R2, R1	10.5 **SBC** IR2, R1	10.5 **SBC** R1, IM	10.5 **SBC** IR1, IM								
4	8.5 **DA** R1	8.5 **DA** IR1	6.5 **OR** r1, r2	6.5 **OR** r1, Ir2	10.5 **OR** R2, R1	10.5 **OR** IR2, R1	10.5 **OR** R1, IM	10.5 **OR** IR1, IM								
5	10.5 **POP** R1	10.5 **POP** IR1	6.5 **AND** r1, r2	6.5 **AND** r1, Ir2	10.5 **AND** R2, R1	10.5 **AND** IR2, R1	10.5 **AND** R1, IM	10.5 **AND** IR1, IM								6.0 **WDT**
6	6.5 **COM** R1	6.5 **COM** IR1	6.5 **TCM** r1, r2	6.5 **TCM** r1, Ir2	10.5 **TCM** R2, R1	10.5 **TCM** IR2, R1	10.5 **TCM** R1, IM	10.5 **TCM** IR1, IM								6.0 **STOP**
7	10/12.1 **PUSH** R2	12/14.1 **PUSH** IR2	6.5 **TM** r1, r2	6.5 **TM** r1, Ir2	10.5 **TM** R2, R1	10.5 **TM** IR2, R1	10.5 **TM** R1, IM	10.5 **TM** IR1, IM								7.0 **HALT**
8	10.5 **DECW** RR1	10.5 **DECW** IR1	12.0 **LDE** r1, Irr2	18.0 **LDEI** Ir1, Irr2												6.1 **DI**
9	6.5 **RL** R1	6.5 **RL** IR1	12.0 **LDE** r2, Irr1	18.0 **LDEI** Ir2, Irr1												6.1 **EI**
A	10.5 **INCW** RR1	10.5 **INCW** IR1	6.5 **CP** r1, r2	6.5 **CP** r1, Ir2	10.5 **CP** R2, R1	10.5 **CP** IR2, R1	10.5 **CP** R1, IM	10.5 **CP** IR1, IM								14.0 **RET**
B	6.5 **CLR** R1	6.5 **CLR** IR1	6.5 **XOR** r1, r2	6.5 **XOR** r1, Ir2	10.5 **XOR** R2, R1	10.5 **XOR** IR2, R1	10.5 **XOR** R1, IM	10.5 **XOR** IR1, IM								16.0 **IRET**
C	6.5 **RRC** R1	6.5 **RRC** IR1	12.0 **LDC** r1, Irr2	18.0 **LDCI** Ir1, Irr2				10.5 **LD** r1,x,R2								6.5 **RCF**
D	6.5 **SRA** R1	6.5 **SRA** IR1	12.0 **LDC** r2, Irr1	18.0 **LDCI** Ir2, Irr1	20.0 **CALL*** IRR1		20.0 **CALL** DA	10.5 **LD** r2,x,R1								6.5 **SCF**
E	6.5 **RR** R1	6.5 **RR** IR1		6.5 **LD** r1, IR2	10.5 **LD** R2, R1	10.5 **LD** IR2, R1	10.5 **LD** R1, IM	10.5 **LD** IR1, IM								6.5 **CCF**
F	8.5 **SWAP** R1	8.5 **SWAP** IR1		6.5 **LD** Ir1, r2		10.5 **LD** R2, IR1										6.0 **NOP**
Bytes per Instruction	2			3					2				3		1	



Legend:
R = 8-bit address
r = 4-bit address
R_1 or r_2 = Dst address
R_1 or r_2 = Src address

Sequence:
Opcode, First Operand, Second Operand

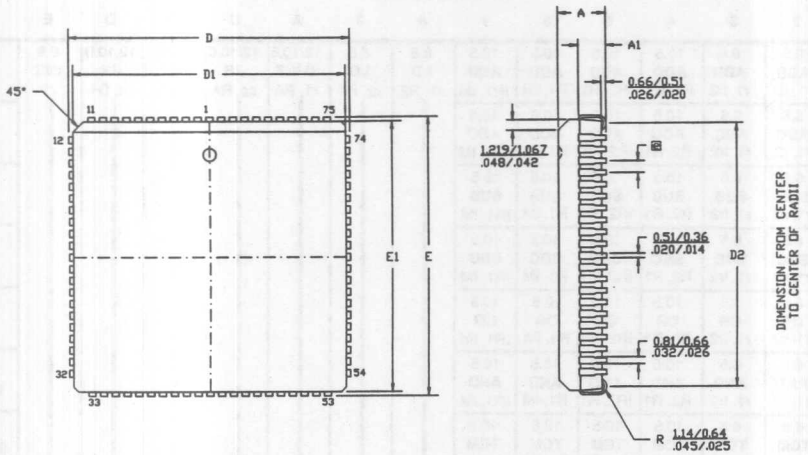
Note: Blank areas not defined.

* 2-byte instruction appears as a 3-byte instruction

† Refer to page 16 for pipeline instructions.

4

PACKAGE INFORMATION



NOTES:
1. CONTROLLING DIMENSIONS : INCH
2. LEADS ARE COPLANAR WITHIN .004 IN.
3. DIMENSION : MM / INCH

SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.32	4.57	.170	.180
A1	2.67	2.92	.105	.115
D/E	30.10	30.35	1.185	1.195
D1/E1	29.21	29.41	1.150	1.158
D2	27.94	28.58	1.100	1.125
e	1.27 TYP		.050 TYP	

84-Pin PLCC Package Diagram

ORDERING INFORMATION

Z89121	Z89921
20 MHz	**20 MHz**
84-Pin PLCC	**84-Pin PLCC**
Z8912120VSC	Z8992120VSC

Codes

Speed
20 = 20.48MHz

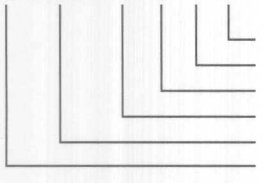
Package
V = Plastic Leaded Chip Carrier (PLCC)

Temperature
S = 0°C to + 70°C

Environment
C = Plastic Standard

Example:

Z 89121 20 V S C is a Z89121, 20.48 MHz, PLCC, 0°C to +70°C, Plastic Standard Flow



4

Z89320

16-BIT DIGITAL SIGNAL PROCESSOR

FEATURES

- 16-Bit Single Cycle Instructions
- Zero Overhead Hardware Looping
- 16-Bit Data
- Ready Control for Slow Peripherals
- Single Cycle Multiply/Accumulate (100 ns)
- Six-Level Stack
- 512 Words of On-Chip RAM
- 16-Bit I/O Port
- 4K Words of On-Chip Masked ROM
- Three Vectored Interrupts
- Two Conditional Branch Inputs/Two User Outputs
- 24-bit ALU, Accumulator and Shifter
- IBM® PC Development Tools
- Cost Effective 40-pin DIP Package

GENERAL DESCRIPTION

The Z89320 is a second generation, 16-bit, fractional, two's complement CMOS Digital Signal Processor (DSP). Most instructions, including multiply and accumulate, are accomplished in a single clock cycle. The processor contains 1 Kbyte of on-chip data RAM (two blocks of 256 16-bit words), 4K words of program ROM. Also, the processor features a 24-bit ALU, a 16 x 16 multiplier, a 24-bit Accumulator and a shifter. Additionally, the processor contains a six-level stack, three vectored interrupts and two inputs for conditional program jumps. Each RAM block contains a set of three pointers which may be incremented or decremented automatically to affect hardware looping without software overhead. The data RAMs can be simultaneously addressed and loaded to the multiplier for a true single cycle multiply.

The device includes a 16-bit I/O bus for transferring data or for mapping peripherals into the processor address space. Additionally, there are two general purpose user inputs and two user outputs. Operation with slow peripherals is accomplished with a ready input pin.

Development tools for the IBM PC include a relocatable assembler, a linker loader, and an ANSI-C compiler. Also, the development tools include a simulator/debugger, a cross assembler for the TMS320 family assembly code and a hardware emulator.

Notes:
All Signals with a preceding front slash, "/", are active Low, e.g., B/W (WORD is active Low); /B/W (BYTE is active Low, only).

Power connections follow conventional descriptions below:

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

5

GENERAL DESCRIPTION (Continued)

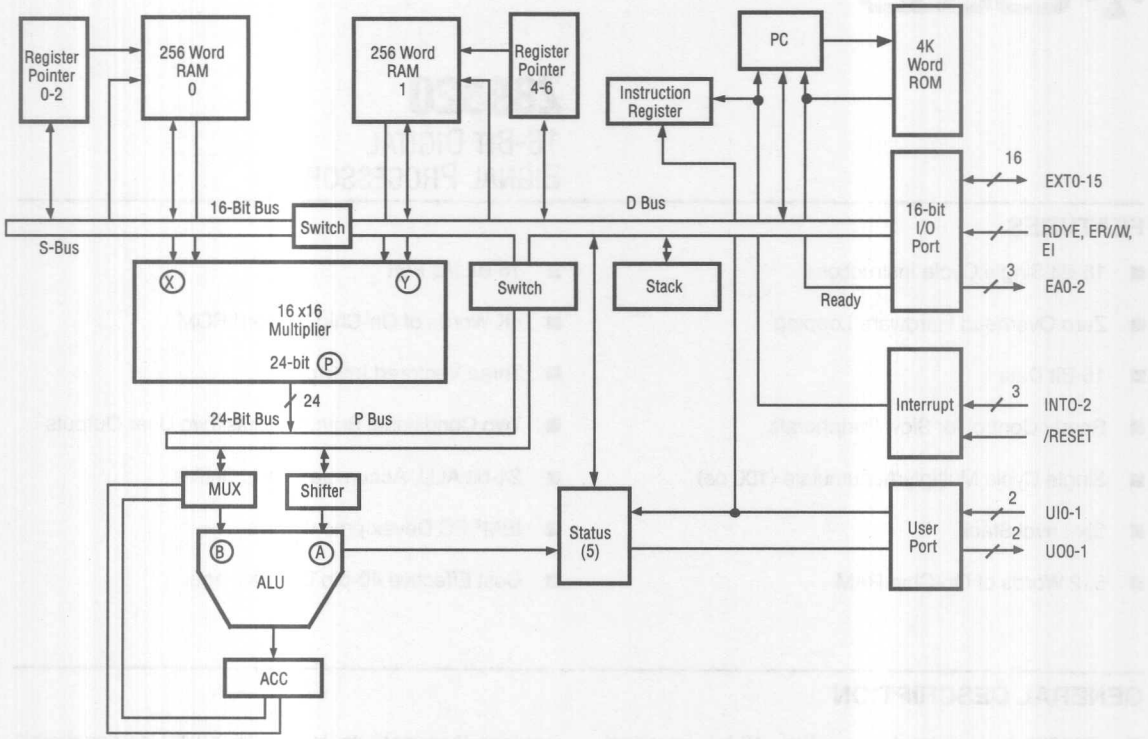


Figure 1. Functional Block Diagram

PIN DESCRIPTION

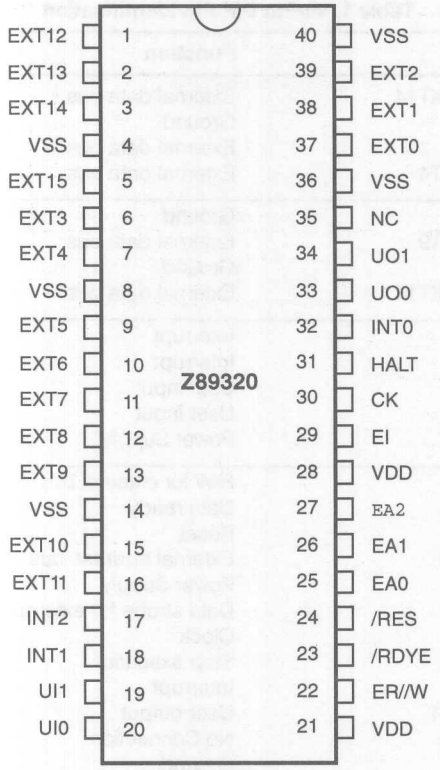


Figure 2. 40-Pin DIP Pin Assignments

5

PIN DESCRIPTION (Continued)

Table 1. 40-Pin DIP Pin Identification

No.	Symbol	Function	Direction
1-3	EXT12-EXT14	External data bus	Input/Output
4	V_{SS}	Ground	Input
5	EXT15	External data bus	Input/Output
6-7	EXT3-EXT4	External data bus	Input/Output
8	V_{SS}	Ground	Input
9-13	EXT5-EXT9	External data bus	Input/Output
14	V_{SS}	Ground	Input
15-16	EXT10-EXT11	External data bus	Input/Output
17	INT2	Interrupt	Input
18	INT1	Interrupt	Input
19	UI1	User input	Input
20	UI0	User input	Input
21	V_{DD}	Power Supply	Input
22	ER/W	R/W for external bus	Output
23	/RDYE	Data ready	Input
24	/RES	Reset	Input
25-27	EA0-EA2	External address bus	Output
28	V_{DD}	Power Supply	Input
29	EI	Data strobe for external bus	Output
30	CK	Clock	Input
31	HALT	Stop execution	Input
32	INT0	Interrupt	Input
33-34	UO0-UO1	User output	Output
35	NC	No Connection	
36	V_{SS}	Ground	Input
37-39	EXT0-EXT2	External data bus	Input/Output
40	V_{SS}	Ground	Input

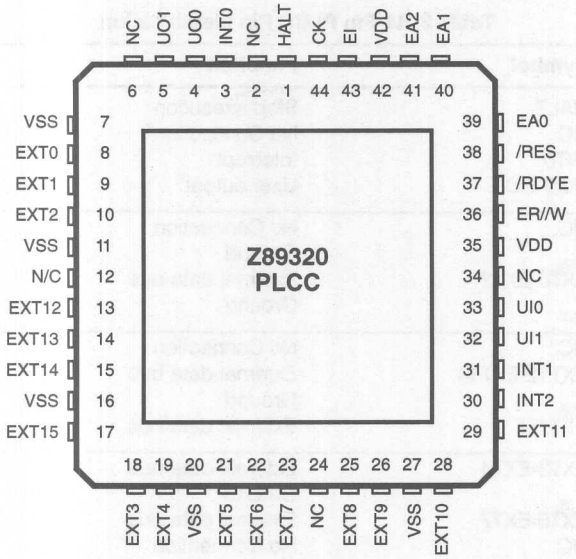


Figure 3. 44-Pin PLCC Pin Assignments (Standard Mode)

PIN DESCRIPTION (Continued)

Table 2. 44-Pin PLCC Pin Identification

No.	Symbol	Function	Direction
1	HALT	Stop execution	Input
2	NC	No Connection	
3	INT0	Interrupt	Input
4-5	UO0-UO1	User output	Output
6	NC	No Connection	
7	V_{SS}	Ground	Input
8-10	EXT0-EXT2	External data bus	Input/Output
11	V_{SS}	Ground	Input
12	NC	No Connection	
13-15	EXT12-EXT14	External data bus	Input/Output
16	V_{SS}	Ground	Input
17	EXT15	External data bus	Input/Output
18-19	EXT3-EXT4	External data bus	Input/Output
20	V_{SS}	Ground	Input
21-23	EXT5-EXT7	External data bus	Input/Output
24	NC	No Connection	
25-26	EXT8-EXT9	External Data Bus	Input/Output
27	V_{SS}	Ground	Input
28-29	EXT10-EXT11	External data bus	Input/Output
30	INT2	Interrupt	Input
31	INT1	Interrupt	Input
32	UI1	User input	Input
33	UI0	User input	Input
34	NC	No Connection	
35	V_{DD}	Power Supply	Input
36	ER/W	R/W for external bus	Output
37	/RDYE	Data ready	Input
38	/RES	Reset	Input
39-41	EA0-EA2	External address bus	Output
42	V_{DD}	Power Supply	Input
43	EI	Data strobe for external bus	Output
44	CK	Clock	Input

PIN FUNCTIONS

CK *Clock* (input). External clock.

EXT15-EXT0 *External Data Bus* (input/output). Data bus for user defined outside registers such as an ADC or DAC. The pins are normally in output mode except when the outside registers are specified as source registers in the instructions. All the control signals exist to allow a read or a write through this bus.

ER/W *External Bus Direction* (output). Data direction signal for EXT-Bus. Data is available from the CPU on EXT15-EXT0 when this signal is Low. EXT-Bus is in input mode (high-impedance) when this signal is High.

EA2-EA0 *External Address* (output). User-defined register address output. One of eight user-defined external registers is selected by the processor with these address pins for read or write operations. Since the addresses are part of the processor memory map, the processor is simply executing internal reads and writes.

EI *Enable Input* (output). Read/Write timing signal for EXT-Bus. User defined register or the processor can put data on the EXT-Bus during a Low state. Data is read by the external peripheral on the rising edge of EI. Data is read by the processor on the rising edge of CK not EI.

HALT *Halt State (input).* Stop Execution Control. The CPU continuously executes NOPs and the program counter remains at the same value when this pin is held High. This signal must be synchronized with CK.

INT2-INT0 *Three Interrupts* (input, active Low). Interrupt request 2-0. Interrupts are generated on the rising edge of the input signal. Interrupt vectors for the interrupt service starting address are stored in the program memory locations 0FFFH for INT0, 0FFEH for INT1, and 0FFDH for INT2. Priority is : INT2 = lowest, INT0 = highest.

/RES *Reset* (input, active Low). Asynchronous reset signal. A Low level on this pin generates an internal reset signal. The /RES signal must be kept Low for at least one clock cycle. The CPU fetches a new Program Counter (PC) value from program memory address 0FFCH after the reset signal is released.

/RDYE *Data Ready* (input). User-supplied Data Ready signal for data to and from external data bus. This pin stretches the EI and ER/W lines and maintains data on the address bus and data bus. The ready signal is sampled at the rising edge of the clock with appropriate setup and hold times. The normal write cycle will continue from the next rising clock only if ready is active (High state).

UI1-UI0 *Two Input Pins* (input). General purpose input pins. These input pins are directly tested by the conditional branch instructions. These are asynchronous input signals that have no special clock synchronization requirements.

UO1-UO0 *Two Output Pins* (output). General purpose output pins. Their value is determined by the status register bits S5 and S6. If a one is loaded into S5 or S6, a Low output appears at the respective pin. If a zero is used, a High output appears.

ADDRESS SPACE

Program Memory. Programs of up to 4K words can be masked into internal ROM. Four locations are dedicated to the vector address for the three interrupts (0FFDH-0FFFH) and the starting address following a Reset (0FFCH). Internal ROM is mapped from 0000H to 0FFFH, and the highest location for program is 0FFBH.

Internal Data RAM. The Z89320 has an internal 512 x 16-bit word data RAM organized as two banks of 256 x 16-bit words each, referred to as RAM0 and RAM1. Each data RAM bank is addressed by three pointers, referred to as Pn:0 (n=0-2) for RAM0 and Pn:1 (n=0-2) for RAM1. The RAM addresses for RAM0 and RAM1 are arranged from 0-255 and 256-511 respectively. The address pointers, which may be written to or read from, are 8-bit registers connected to the lower byte of the internal 16-bit D-Bus and are used to perform no overhead looping. Three addressing modes are available to access the Data RAM: register indirect, direct addressing, and short form direct. These modes are discussed in detail later. The contents of the RAM can be read or written in one machine cycle per word without disturbing any internal registers or status other than the RAM address pointer used for each RAM. The contents of each RAM can be loaded simultaneously into the X and Y inputs of the multiplier.

Registers. The Z89320 has 12 internal registers and up to an additional eight external registers. The external registers are user definable for peripherals such as A/D or D/A or to DMA or other addressing peripherals. External registers are accessed in one machine cycle the same as internal registers.

5

FUNCTIONAL DESCRIPTION

General. The Z89320 is a high-performance Digital Signal Processor with a modified Harvard-type architecture with separate program and data memory. The design has been optimized for processing power and minimizing silicon space.

Instruction Timing. Many instructions are executed in one machine cycle. Long immediate instructions and Jump or Call instructions are executed in two machine cycles. When the program memory is referenced in internal RAM indirect mode, it takes three machine cycles. In addition, one more machine cycle is required if the PC is selected as the destination of a data transfer instruction. This only happens in the case of a register indirect branch instruction.

An Acc + P => Acc; a(i) * b(j) → P calculation and modification of the RAM pointers, is done in one machine cycle. Both operands, a(i) and b(j), can be located in two independent RAM (0 and 1) addresses.

Multiply/Accumulate. The multiplier can perform a 16-bit x 16-bit multiply or multiply accumulate in one machine cycle using the Accumulator and/or both the X and Y inputs. The multiplier produces a 32-bit result, however, only the 24 most significant bits are saved for the next instruction or accumulation. For operations on very small numbers where the least significant bits are important, the data should first be scaled by eight bits (or the multiplier and multiplicand by four bits each) to avoid truncation errors. Note that all inputs to the multiplier should be fractional two's complement 16-bit binary numbers. This puts them in the range [–1 to 0.9999695], and the result is in 24 bits so that the range is [–1 to 0.9999999]. In addition, if 8000H is loaded into both X and Y registers, the resulting multiplication is considered an illegal operation as an overflow would result. Positive one cannot be represented in fractional notation, and the multiplier will actually yield the result 8000H x 8000H = 8000H (–1 x –1 = –1).

ALU. The 24-bit ALU has two input ports, one of which is connected to the output of the 24-bit Accumulator. The other input is connected to the 24-bit P-Bus, the upper 16 bits of which are connected to the 16-bit D-Bus. A shifter between the P-Bus and the ALU input port can shift the data by three bits right, one bit right, one bit left or no shift.

Hardware Stack. A six-level hardware stack is connected to the D-Bus to hold subroutine return addresses or data. The Call instruction pushes PC+2 onto the stack. The RET instruction pops the contents of the stack to the PC.

User Inputs. The Z89320 has two inputs, UI0 and UI1, which may be used by Jump and Call instructions. The Jump or Call tests one of these pins and if appropriate, jumps to a new location. Otherwise, the instruction behaves like a NOP. These inputs are also connected to the status register bits S10 and S11 which may be read by the appropriate instruction (Figure 4).

User Outputs. The status register bits S5 and S6 are connected to UO0 and UO1 pins and may be written to by the appropriate instruction. The status bits are inverted prior to being output to the external pin.

Interrupts. The Z89320 has three positive edge-triggered interrupt inputs. An interrupt is acknowledged at the end of any instruction execution. It takes two machine cycles to enter an interrupt instruction sequence. The PC is pushed onto the stack. A RET instruction transfers the contents of the stack to the PC and decrements the stack pointer by one word. The priority of the interrupts is INT0 = highest, INT2 = lowest.

Registers. The Z89320 has 12 physical internal registers and up to eight user-defined external registers. The EA2-EA0 determines the address of the external registers. The /EI, /RDYE, and ER//W signals are used to read or write from the external registers.

REGISTERS

There are 12 internal registers which are defined below:

Register	Register Definition
P	Output of Multiplier, 24-Bit
X	X Multiplier Input, 16-Bit
Y	Y Multiplier Input, 16-Bit
A	Accumulator, 24-Bit
SR	Status Register, 16-Bit
Pn:b	Six Ram Address Pointers, 8-Bit Each
PC	Program Counter, 16-Bit

The following are virtual registers as physical RAM does not exist on the chip.

EXTn	External registers, 16-bit
BUS	D-Bus
Dn:b	Eight Data Pointers

P holds the result of multiplications and is read-only.

X and **Y** are two 16-bit input registers for the multiplier. These registers can be utilized as temporary registers when the multiplier is not being used. Since the multiplier provides a flow through process, any data placed in the X or Y register automatically invokes a multiplication.

A is a 24-bit Accumulator. The output of the ALU is sent to this register. When 16-bit data is transferred into this register, it goes into the 16 MSB's and the least significant eight bits are set to zero. Only the upper 16 bits are transferred to the destination register when the Accumulator is selected as a source register in transfer instructions.

Pn:b are the pointer registers for accessing data RAM. (n=0,1,2 refer to the pointer number) (b=0,1 refers to RAM Bank 0 or 1). They can be directly read from or written to, and can point to locations in data RAM or Program Memory.

EXTn are external registers (n=0 to 7). There are eight 16-bit registers here for mapping external devices into the address space of the processor. Note that the actual register RAM does not exist on the chip, but would exist as part of the external device such as an ADC result latch.

bus is a read-only register which, when accessed, returns the contents of the D-Bus.

Dn:b refer to possible locations in RAM that can be used as a pointer to locations in program memory. The programmer decides which location to choose from two bits in the status register and two bits in the operand. Thus, only the lower 16 possible locations in RAM can be specified. At any one time there are eight usable pointers, four per bank, and the four pointers are in consecutive locations in RAM. For example, if S3/S4 = 01 in the status register, then D0:0/D1:0/D2:0/D3:0 refer to locations 4/5/6/7 in RAM Bank 0. Note that when the data pointers are being written to, a number is actually being loaded to Data RAM, so they can be used as a limited method for writing to RAM.

REGISTERS (Continued)

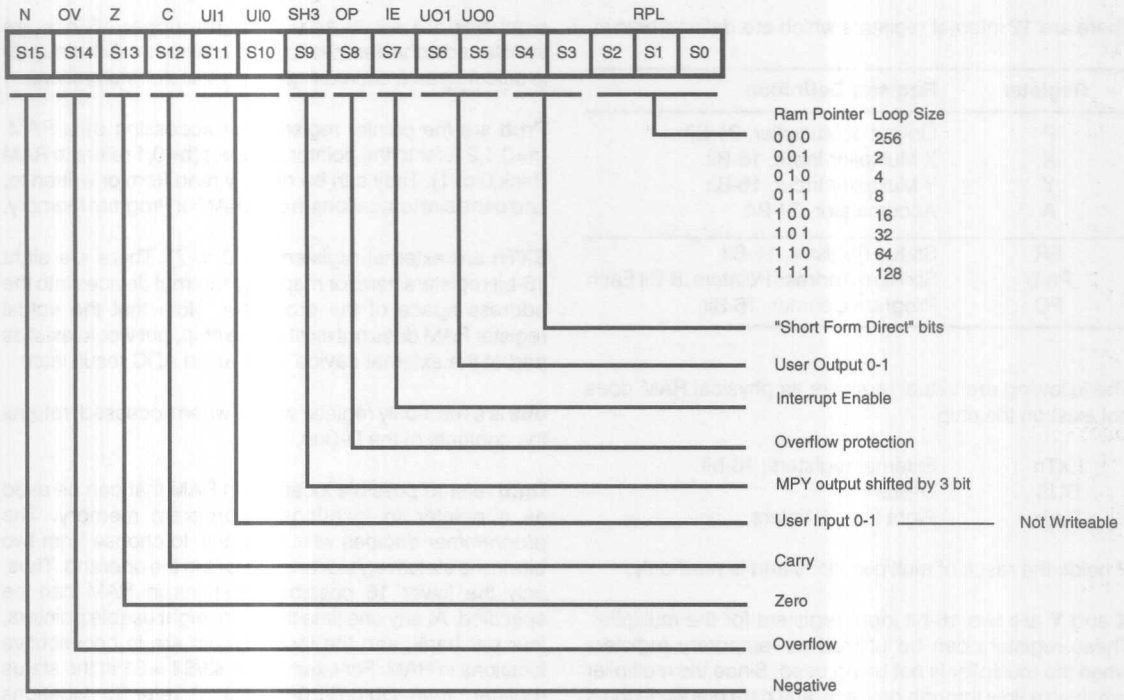


Figure 4. Status Register

SR is the status register (Figure 4) which contains the ALU status and certain control bits as shown in the following table.

Table 3. Status Register Bit Functions

Status Register Bit	Function
S15 (N)	ALU Negative
S14 (OV)	ALU Overflow
S13 (Z)	ALU Zero
S12 (L)	Carry
S11 (UI1)	User Input 1
S10 (UI0)	User Input 0
S9 (SH3)	MPY Output Shifted by 3 Bits
S8 (OP)	Overflow Protection
S7 (IE)	Interrupt Enable
S6 (UO1)	User Output 1
S5 (UO0)	User Output 0
S4-S3	"Short Form Direct" Bits
S2-S0 (RPL)	RAM Pointer Loop Size

Table 4. RPL Description

S2	S1	S0	Loop Size
0	0	0	256
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The status register may always be read in its entirety. S15-S10 are set/reset by the hardware and can only be read by software. S9-S0 can be written by software.

S15-S12 are set/reset by the ALU after an operation. S11-S10 are set/reset by the user inputs. S6-S0 are control bits described elsewhere. S7 enables interrupts. S8, if 0 (reset), allows the hardware to overflow. If S8 is set, the hardware clamps at maximum positive or negative values instead of overflowing.

If S9 is set and the MPYA or MPYS instruction is used, then the shifter to the ALU shifts the result three bits right.

PC is the Program Counter. When this register is assigned as a destination register, one NOP machine cycle is added automatically to adjust the pipeline timing.

RAM ADDRESSING

The address of the RAM is specified in one of three ways (Figure 5):

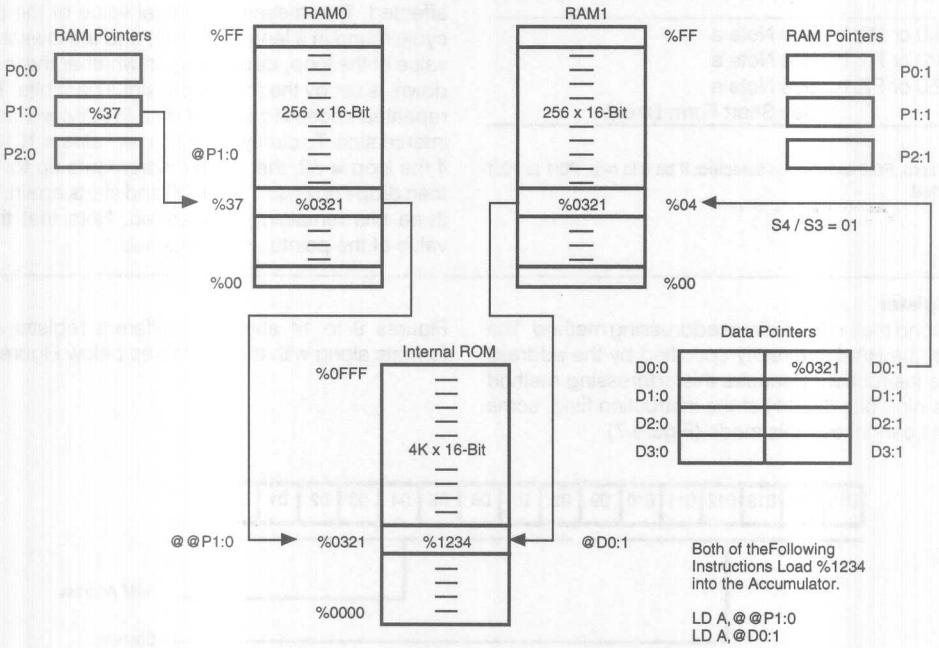


Figure 5. RAM, ROM, and Pointer Architecture

5

Register Indirect

Pn:b n=0-2, b=0-1

The most commonly used method is a register indirect addressing method, where the RAM address is specified by one of the three RAM address pointers (n) for each bank (b). Each source/destination field in Figures 6 and 9 may be used by an indirect instruction to specify a register pointer and its modification after execution of the instruction.

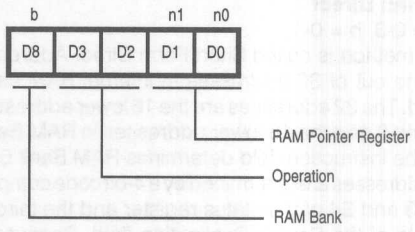


Figure 6. Indirect Register

RAM ADDRESSING (Continued)

The register pointer is specified by the first and second bits in the source/destination field and the modification is specified by the third and fourth bits according to the following table:

D3-D0		Meaning
00xx	NOP	No Operation
01xx	+1	Simple Increment
10xx	–1/LOOP	Decrement Modulo the Loop Count
11xx	+1/LOOP	Increment Modulo the Loop Count
xx00	P0:0 or P0:1	See Note a
xx01	P1:0 or P1:1	See Note a
xx10	P2:0 or P2:1	See Note a
xx11		See Short Form Direct

Notes:

a. If bit 8 is zero, P0:0 to P2:0 are selected; if bit 8 is one, P0:1 to P2:1 are selected.

When LOOP mode is selected, the pointer to which the loop is referring will cycle up or down, depending on whether a –LOOP or +LOOP is specified. The size of the loop is obtained from the least significant three bits of the Status Register. The increment or decrement of the register is accomplished modulo the loop size. As an example, if the loop size is specified as 32 by entering the value 101 into bits 2-0 of the Status Register (S2-S0) and an increment +LOOP is specified in the address field of the instruction, i.e., the RPi field is 11xx, then the register specified by RPi will increment, but only the least significant five bits will be affected. This means the actual value of the pointer will cycle round in a length 32 loop, and the lowest or highest value of the loop, depending on whether the loop is up or down, is set by the three most significant bits. This allows repeated access to a set of data in RAM without software intervention. To clarify, if the pointer value is 10101001 and if the loop = 32, the pointer increments up to 10111111, then drops down to 10100000 and starts again. The upper three bits remaining unchanged. Note that the original value of the pointer is not retained.

Direct Register

The second method is a direct addressing method. The address of the RAM is directly specified by the address field of the instruction. Because this addressing method consumes nine bits (0-511) of the instruction field, some instructions cannot use this mode (Figure 7).

Figures 9 to 14 show the different register instruction formats along with the two tables below Figure 9.

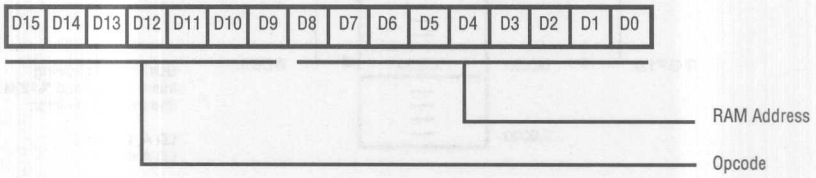


Figure 7. Direct Internal RAM Address Format

Short Form Direct

Dn:b n = 0-3, b = 0-1

The last method is called Short Form Direct Addressing, where one out of 32 addresses in internal RAM can be specified. The 32 addresses are the 16 lower addresses in RAM Bank 0 and the 16 lower addresses in RAM Bank 1. Bit 8 of the instruction field determines RAM Bank 0 or 1. The 16 addresses are determined by a 4-bit code comprised of bits S3 and S4 of the status register and the third and fourth bits of the Source/Destination field. Because this mode can specify a direct address in a short form, all of the instructions using the register indirect mode can use this mode (Figure 8). This method can access only the lower 16 addresses in the both RAM banks and as such has limited use. The main purpose is to specify a data register, located in the RAM bank, which can then be used to point to a program memory location. This facilitates down-loading look-up tables etc. from program memory to RAM.

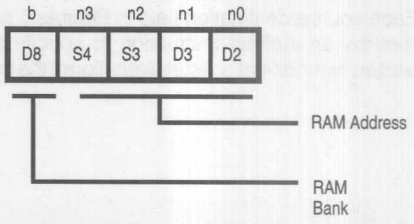
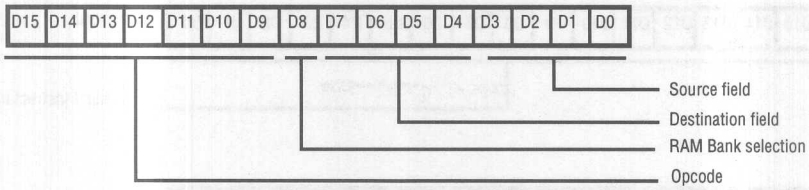


Figure 8. Short Form Direct Address

INSTRUCTION FORMAT



Note:
Source/Destination fields can specify either register or RAM addresses in RAM pointer indirect mode.

Figure 9. General Instruction Format

Table 5. Registers

Source/Destination	Register
0000	BUS**
0001	X
0010	Y
0011	A
0100	SR
0101	STACK
0110	PC
0111	P**
1000	EXT0
1001	EXT1
1010	EXT2
1011	EXT3
1100	EXT4
1101	EXT5
1110	EXT6
1111	EXT7

Table 6. Register Pointers Field

Source/Destination	Meaning
00xx	NOP
01xx	+1
10xx	–1/LOOP
11xx	+1/LOOP
xx00	P0:0 or P0:1*
xx01	P1:0 or P1:1*
xx10	P2:0 or P2:1*
xx11	Short Form Direct Mode***

Notes:
* If RAM Bank bit is 0, then Pn:0 are selected.
If RAM Bank bit is 1, then Pn:1 are selected.
** Read only.
*** When the short form direct mode is selected, 00000-01111 or 10000-11111 are used as RAM addresses.

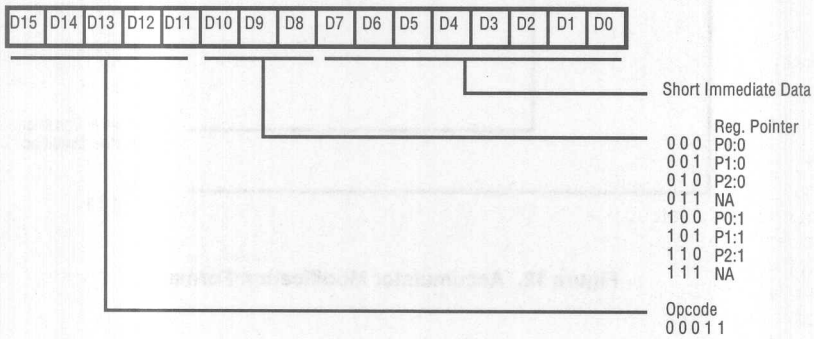


Figure 10. Short Immediate Data Load Format

5

INSTRUCTION FORMAT (Continued)

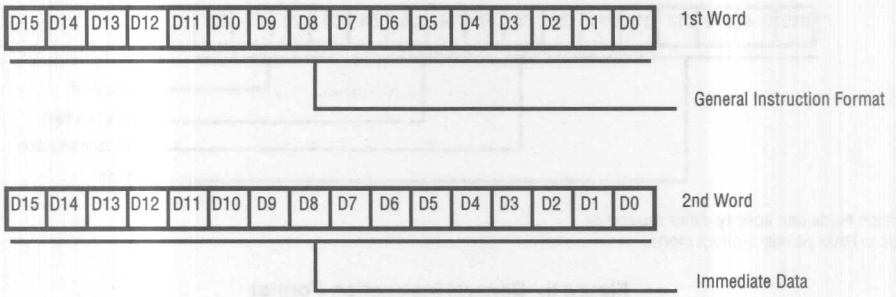


Figure 11. Immediate Data Load Format

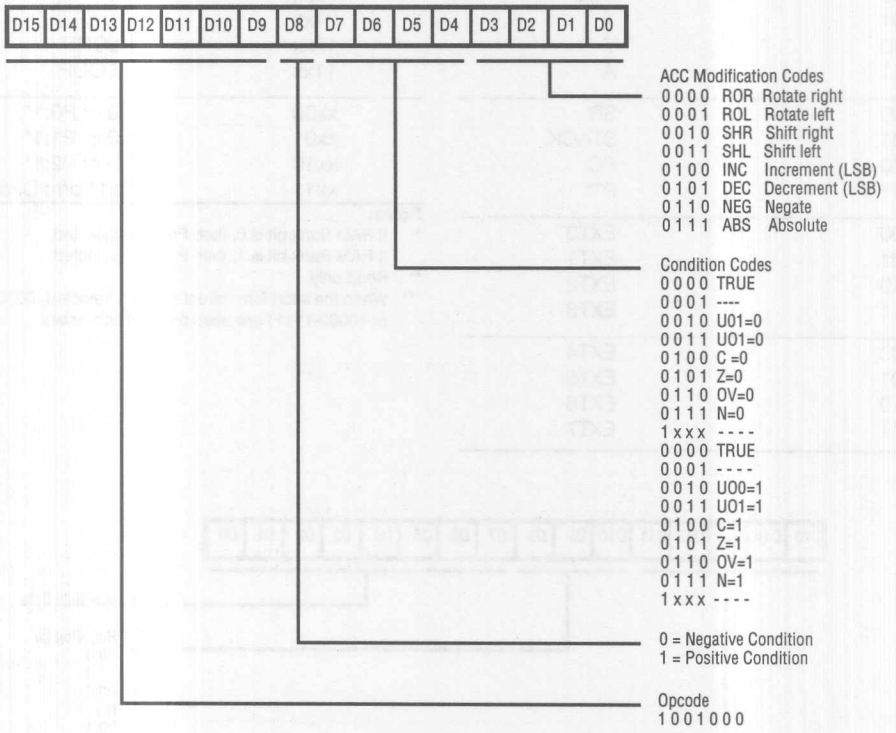


Figure 12. Accumulator Modification Format

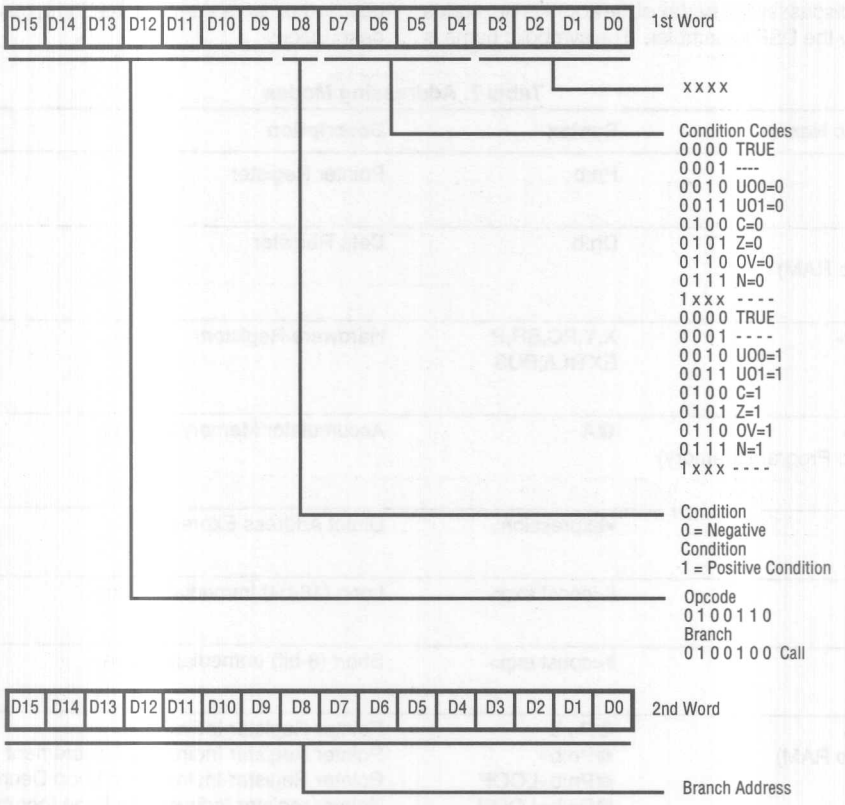


Figure 13. Branching Format

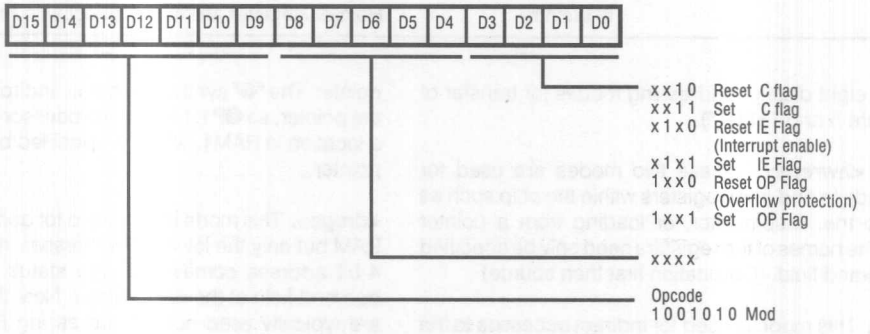


Figure 14. Flag Modification Format

5

ADDRESSING MODES

This section discusses the syntax of the addressing modes supported by the DSP assembler. The symbolic name is used in the discussion of instruction syntax in the instruction descriptions.

Table 7. Addressing Modes

Symbolic Name	Syntax	Description
<pregs>	Pn:b	Pointer Register
<dregs> (Points to RAM)	Dn:b	Data Register
<hwregs>	X,Y,PC,SR,P EXTn,A,BUS	Hardware Registers
<accind> (Points to Program Memory)	@A	Accumulator Memory Indirect
<direct>	<expression>	Direct Address Expression
<limm>	#<const exp>	Long (16-bit) Immediate Value
<simm>	#<const exp>	Short (8-bit) Immediate Value
<regind> (Points to RAM)	@Pn:b @Pn:b+ @Pn:b–LOOP @Pn:b+LOOP	Pointer Register Indirect Pointer Register Indirect with Increment Pointer Register Indirect with Loop Decrement Pointer register Indirect with Loop Increment
<memind> (Points to Program Memory)	@@Pn:b @Dn:b @@Pn:b–LOOP @@Pn:b+LOOP @@Pn:b+	Pointer Register Memory Indirect Data Register Memory Indirect Pointer Register Memory Indirect with Loop Decrement Pointer Register Memory Indirect with Loop Increment Pointer Register Memory Indirect with Increment

There are eight distinct addressing modes for transfer of data (Figure 5 and Table 7).

<pregs>, <hwregs>. These two modes are used for simple loads to and from registers within the chip such as loading to the Accumulator, or loading from a pointer register. The names of the registers need only be specified in the operand field. (Destination first then source)

<regind>. This mode is used for indirect accesses to the data RAM. The address of the RAM location is stored in the pointer. The "@" symbol indicates "indirect" and precedes the pointer, so @P1:1 tells the processor to read or write to a location in RAM1, which is specified by the value in the pointer.

<dregs>. This mode is also used for accesses to the data RAM but only the lower 16 addresses in either bank. The 4-bit address comes from the status register and the operand field of the data pointer. Note that data registers are typically used not for addressing RAM, but loading data from program memory space.

<memind>. This mode is used for indirect, indirect accesses to the program memory. The address of the memory is located in a RAM location, which is specified by the value in a pointer. So @@P1:1 tells the processor to read (write is not possible) from a location in memory, which is specified by a value in RAM, and the location of the RAM is in turn specified by the value in the pointer. Note that the data pointer can also be used for a memory access in this manner, but only one "@" precedes the pointer. In both cases the memory address stored in RAM is incremented by one each time the addressing mode is used to allow easy transfer of sequential data from program memory.

<accind>. Similar to the previous mode, the address for the program memory read is stored in the Accumulator. @A in the second operand field loads the number in memory specified by the address in A.

<direct>. The direct mode allows read or write to data RAM from the Accumulator by specifying the absolute address of the RAM in the operand of the instruction. A number between 0 and 255 indicates a location in RAM0, and a number between 256 and 511 indicates a location in RAM1.

<limm>. This indicates a long immediate load. A 16-bit word can be copied directly from the operand into the specified register or memory.

<simm>. This can only be used for immediate transfer of 8-bit data in the operand to the specified RAM pointer.

CONDITION CODES

The following defines the condition codes supported by the DSP assembler. If the instruction description refers to the <cc> (condition code) symbol in one of its addressing modes, the instruction will only execute if the condition is true.

Table 8. Condition Codes

Name	Description	Name	Description
C	Carry	NU1	Not User One
EQ	Equal (same as Z)	NZ	Not zero
F	False	OV	Overflow
IE	Interrupts Enabled	PL	Plus (Positive)
MI	Minus	U0	User Zero
NC	No Carry	U1	User One
NE	Not Equal (same as NZ)	UGE	Unsigned Greater Than or Equal (Same as NC)
NIE	Not Interrupts Enabled		
NOV	Not Overflow	ULT	Unsigned Less Than (Same as C)
NU0	Not User Zero	Z	Zero

5

INSTRUCTION DESCRIPTIONS

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
ABS	Absolute Value	ABS[<cc>,]<src>	<cc>,A	1	1	ABS NC,A
			A	1	1	ABS A
ADD	Addition	ADD<dest>,<src>	A,<pregs>	1	1	ADD A,P0:0
			A,<dregs>	1	1	ADD A,D0:0
			A,<limm>	2	2	ADD A,#%1234
			A,<memind>	1	3	ADD A,@@P0:0
			A,<direct>	1	1	ADD A,%F2
			A,<regind>	1	1	ADD A,@P1:1
			A,<hwregs>	1	1	ADD A,X
AND	Bitwise AND	AND<dest>,<src>	A,<pregs>	1	1	AND A,P2:0
			A,<dregs>	1	1	AND A,D0:1
			A,<limm>	2	2	AND A,#%1234
			A,<memind>	1	3	AND A,@@P1:0
			A,<direct>	1	1	AND A,%2C
			A,<regind>	1	1	AND A,@P1:2+LOOP
			A,<hwregs>	1	1	AND A,EXT3
CALL	Subroutine call	CALL [<cc>,]<address>	<cc>,<direct>	2	2	CALL Z,sub2
			<direct>	2	2	CALL sub1
CCF	Clear carry flag	CCF	None	1	1	CCF
CIEF	Clear Carry Flag	CIEF	None	1	1	CIEF
COPF	Clear OP flag	COPF	None	1	1	COPF
CP	Comparison	CP<src1>,<src2>	A,<pregs>	1	1	CP A,P0:0
			A,<dregs>	1	1	CP A,D3:1
			A,<memind>	1	3	CP A,@@P0:1
			A,<direct>	1	1	CP A,%FF
			A,<regind>	1	1	CP A,@P2:1+
			A,<hwregs>	1	1	CP A,STACK
			A<limm>	2	2	CP A,#%FFCF
DEC	Decrement	DEC [<cc>,]<dest>	<cc>A,	1	1	DEC NZ,A
			A	1	1	DEC A
INC	Increment	INC [<cc>,] <dest>	<cc>,A	1	1	INC PL,A
			A	1	1	INC A
JP	Jump	JP [<cc>,]<address>	<cc>,<direct>	2	2	JP NIE,Label
			<direct>	2	2	JP Label

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
LD	Load destination with source	LD<dest>,<src>	A,<hwregs>	1	1	LD A,X
			A,<dregs>	1	1	LD A,D0:0
			A,<pregs>	1	1	LD A,P0:1
			A,<regind>	1	1	LD A,@P1:1
			A,<memind>	1	3	LD A,@D0:0
			A,<direct>	1	1	LD A,124
			<direct>,A	1	1	LD 124,A
			<dregs>,<hwregs>	1	1	LD D0:0,EXT7
			<pregs>,<simm>	1	1	LD P1:1,#%FA
			<pregs>,<hwregs>	1	1	LD P1:1,EXT1
			<regind>,<limm>	1	1	LD@P1:1,#1234
			<regind>,<hwregs>	1	1	LD @P1:1+,X
			<hwregs>,<pregs>	1	1	LD Y,P0:0
			<hwregs>,<dregs>	1	1	LD SR,D0:0
			<hwregs>,<limm>	2	2	LD PC,#%1234
			<hwregs>,<accind>	1	3	LD X,@A
			<hwregs>,<memind>	1	3	LD Y,@D0:0
			<hwregs>,<regind>	1	1	LD A,@P0:0-LOOP
			<hwregs>,<hwregs>	1	1	LD X,EXT6
			Note: When <dest> is <hwregs>, <dest> cannot be P. Note: When <dest> is <hwregs> and <src> is <hwregs>, <dest> cannot be EXTn if <src> is EXTn, <dest> cannot be X if <src> is X, <dest> cannot be SR if <src> is SR. Note: When <src> is <accind> <dest> cannot be A.			
MLD	Multiply	MLD<srcl>,<srcl>[,<bank switch>]	<hwregs>,<regind>	1	1	MLD A,@P0:0+LOOP
			<hwregs>,<regind>,<bank switch>	1	1	MLD A,@P1:0,OFF
			<regind>,<regind>	1	1	MLD @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MLD @P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <band switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			
MPYA	Multiply and add	MPYA <srcl>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYA A,@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYA A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYA @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYA@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			

INSTRUCTION DESCRIPTIONS (Continued)

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
MPYS	Multiply and subtract	MPYS<src1>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYS A,@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYS A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYS @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYS@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, <regind> the <bank switch> defaults to ON.			
NEG	Negate	NEG <cc>,A	<cc>, A	1	1	NEG MI,A
			A	1	1	NEG A
NOP	No operation	NOP	None	1	1	NOP
OR	Bitwise OR	OR <dest>,<src>	A, <pregs>	1	1	OR A,P0:1
			A, <dregs>	1	1	OR A, D0:1
			A, <limm>	2	2	OR A,#%2C21
			A, <memind>	1	3	OR A,@@P2:1+
			A, <direct>	1	1	OR A, %2C
			A, <regind>	1	1	OR A,@P1:0-LOOP
			A, <hwregs>	1	1	OR A,EXT6
POP	Pop value from stack	POP <dest>	<pregs>	1	1	POP P0:0
			<dregs>	1	1	POP D0:1
			<regind>	1	1	POP @P0:0
			<hwregs>	1	1	POP A
PUSH	Push value onto stack	PUSH <src>	<pregs>	1	1	PUSH P0:0
			<dregs>	1	1	PUSH D0:1
			<regind>	1	1	PUSH @P0:0
			<hwregs>	1	1	PUSH BUS
			<limm>	2	2	PUSH #12345
			<accind>	1	3	PUSH @A
			<memind>	1	3	PUSH @@P0:0
RET	Return from subroutine	RET	None	1	2	RET
RL	Rotate Left	RL <cc>,A	<cc>,A	1	1	RL NZ,A
			A	1	1	RL A
RR	Rotate Right	RR <cc>,A	<cc>,A	1	1	RR C,A
			A	1	1	RR A

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
SCF	Set C flag	SCF	None	1	1	SCF
SIEF	Set IE flag	SIEF	None	1	1	SIEF
SLL	Shift left logical	SLL	[<cc>,]A	1	1	SLL NZ,A
			A	1	1	SLL A
SOPF	Set OP flag	SOPF	None	1	1	SOPF
SRA	Shift right arithmetic	SRA<cc>,A	<cc>,A	1	1	SRA NZ,A
			A	1	1	SRA A
SUB	Subtract	SUB<dest>,<src>	A,<pregs>	1	1	SUB A,P1:1
			A,<dregs>	1	1	SUB A,D0:1
			A,<limm>	2	2	SUB A,#%2C2C
			A, <memind>	1	3	SUB A,@D0:1
			A, <direct>	1	1	SUB A,%15
			A, <regind>	1	1	SUB A,@P2:0-LOOP
			A, <hwregs>	1	1	SUB A,STACK
XOR	Bitwise exclusive OR	XOR <dest>,<src>	A, <pregs>	1	1	XOR A,P2:0
			A, <dregs>	1	1	XOR A,D0:1
			A, <limm>	2	2	XOR A,#13933
			A, <memind>	1	3	XOR A,@@P2:1+
			A, <direct>	1	1	XOR A,%2F
			A, <regind>	1	1	XOR A,@P2:0
			A, <hwregs>	1	1	XOR A,BUS

Bank Switch Enumerations. The third (optional) operand of the MLD, MPYA and MPYS instructions represents whether a bank switch is set on or off. To more clearly represent this two keywords are used (ON and OFF) which state the direction of the switch. These keywords are referred to in the instruction descriptions through the <bank switch> symbol.

5

ABSOLUTE MAXIMUM RATINGS

Symbol	Description	Min	Max	Units
V_{CC}	Supply Voltage (*)	−0.3	+7.0	V
T_{STG}	Storage Temp	−65°	+150°	C
T_A	Oper Ambient Temp		†	C

Notes:
* Voltage on all pins with respect to GND.
† See Ordering Information.

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended period may affect device reliability.

STANDARD TEST CONDITIONS

The characteristics listed below apply for standard test conditions as noted. All voltages are referenced to ground. Positive current flows into the referenced pin (Figure 15).

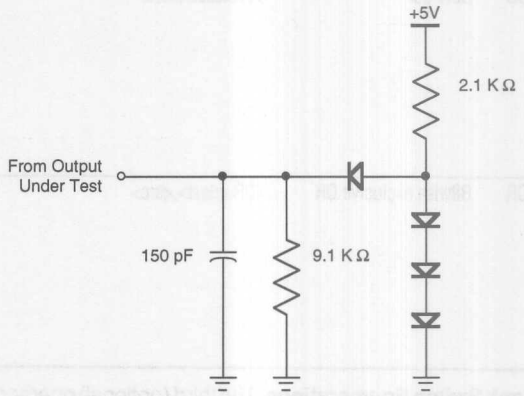


Figure 15. Test Load Diagram

DC ELECTRICAL CHARACTERISTICS

(V_{DD}= 5V ±5%, T_A = 0°C to +70°C unless otherwise specified)

Symbol	Parameter	Condition	Min.	Max.	Units
I_{DD}	Supply Current	V_{DD} = 5.25V fclock = 10 MHz		60	mA
I_{DC}	DC Power Consumption	V_{DD} = 5.25V	1	5	mA
V_{IH}	Input High Level		0.9 V_{DD}		V
V_{IL}	Input Low Level			0.1 V_{DD}	V
IL	Input Leakage			1	µA
V_{OH}	Output High Voltage	I_{OH} = −100 µA	V_{DD}−0.2		V
V_{OL}	Output Low Voltage	I_{OL} = 0.5 mA		0.5	V
I_{FL}	Output Floating Leakage Current			5	µA

AC ELECTRICAL CHARACTERISTICS

(V_{DD} = 5V ±5%, T_A = 0°C to +70°C unless otherwise specified)

Symbol	Parameter	Min.	Max.	Units
TCY	Clock Cycle Time	100	1000	ns
PWW	Clock Pulse Width	45		ns
Tr	Clock Rise Time*	2	4	ns
Tf	Clock Fall Time*	2	4	ns
TEAD	EA,ER/W Delay from CK	15	25	ns
TXVD	EXT Data Output Valid from CK	5	25	ns
TXWH	EXT Data Output Hold from CK	15		ns
TXRS	EXT Data Input Setup Time	15		ns
TXRH	EXT Data Input Hold from CK	0	15	ns
TIED	/EI Delay Time from CK	0	5	ns
RDYS	Ready Setup Time	10		ns
RDYH	Ready Hold Time	0		ns

5

AC TIMING DIAGRAM

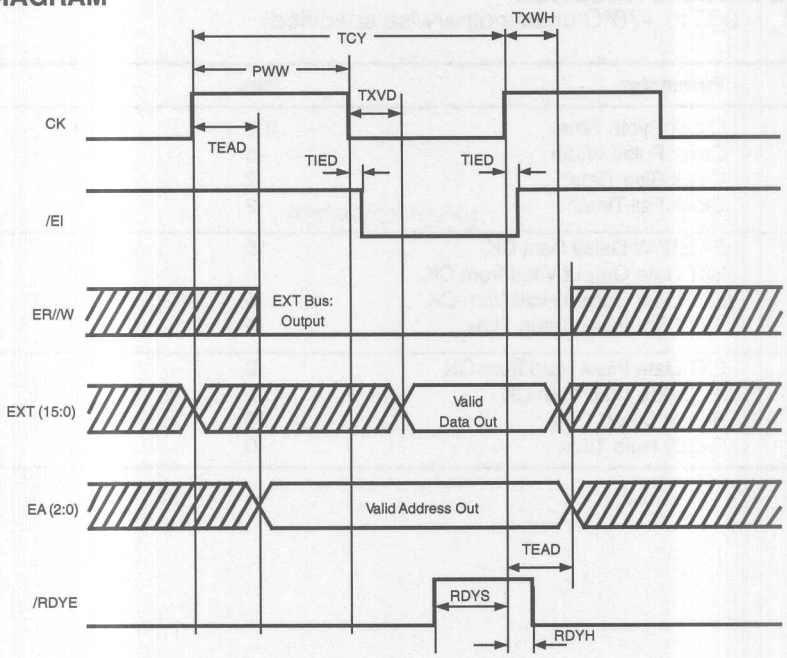


Figure 16. WRITE to external device timing

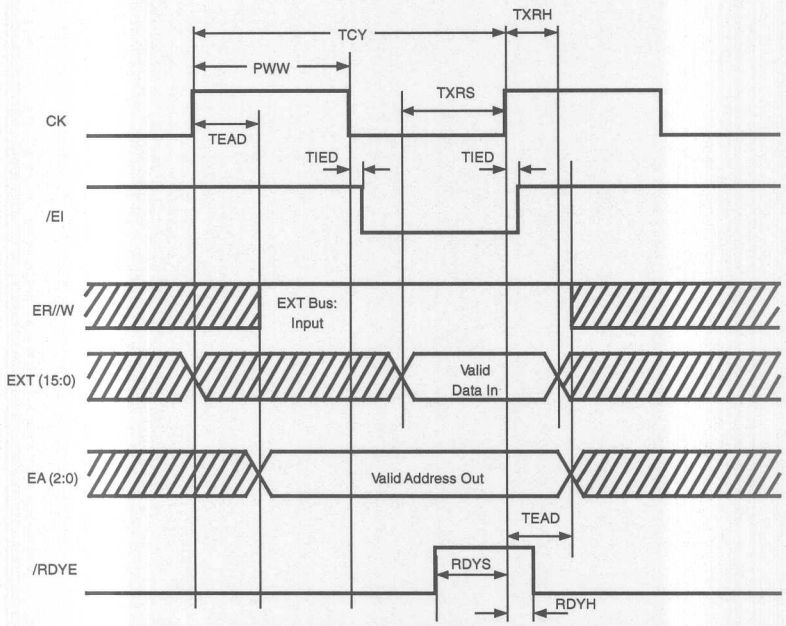
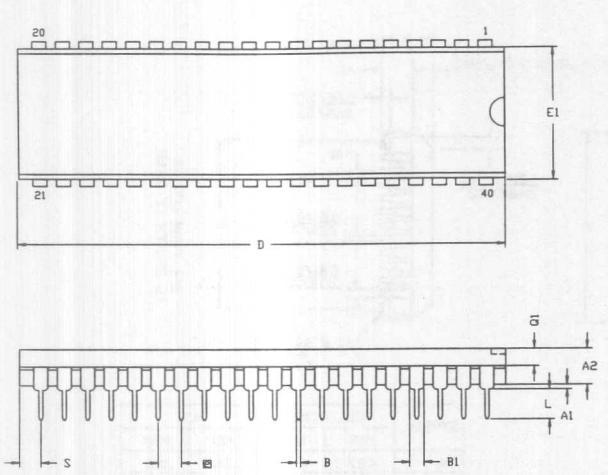


Figure 17. READ from external device timing

PACKAGE INFORMATION



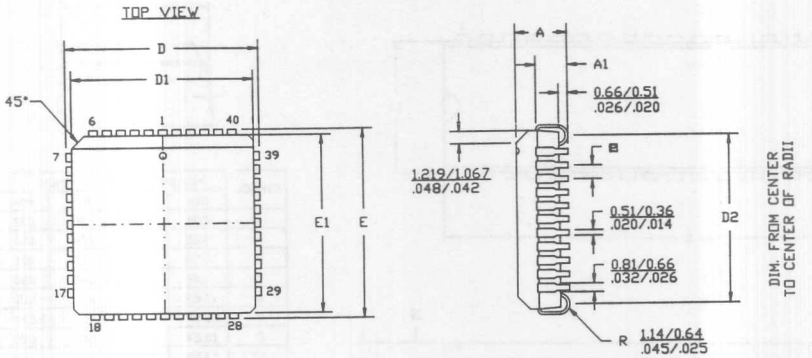
SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A1	0.51	0.81	.020	.032
A2	3.25	3.43	.128	.135
B	0.38	0.53	.015	.021
B1	1.02	1.52	.040	.060
C	0.23	0.38	.009	.015
D	52.07	52.58	2.050	2.070
E	15.24	15.75	.600	.620
E1	13.59	14.22	.535	.560
e	2.54 TYP		.100 TYP	
eA	15.49	16.51	.610	.650
L	3.18	3.81	.125	.150
Q1	1.52	1.91	.060	.075
S	1.52	2.29	.060	.090

CONTROLLING DIMENSIONS : INCH

40-Pin DIP Package Diagram

5

PACKAGE INFORMATION (Continued)



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.27	4.57	.168	.180
A1	2.67	2.92	.105	.115
D/E	17.40	17.65	.685	.695
D1/E1	16.51	16.66	.650	.656
D2	15.24	16.00	.600	.630
e	1.27 TYP		.050 TYP	

NOTES:
1. CONTROLLING DIMENSIONS : INCH
2. LEADS ARE COPLANAR WITHIN .004 IN.
3. DIMENSION : MM / INCH

44-Pin PLCC Package Diagram

ORDERING INFORMATION

Z89320

10 MHz	10 MHz
40-pin DIP	44-pin PLCC
Z8932010PSC	Z8932010VSC

For fast results, contact your local Zilog sales office for assistance in ordering the part desired.

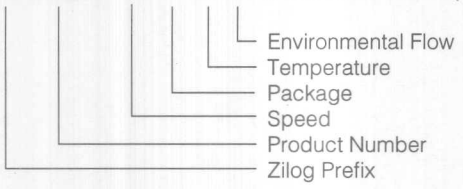
Package
P = Plastic DIP
V = Plastic Chip Carrier

Temperature
S = 0°C to +70°C

Speed
10 = 10 MHz

Environmental
C = Plastic Standard

Example:
Z Z89320 10 P S C is a Z89320 10 MHz, DIP, 0°C to +70°C, Plastic Standard Flow.



5

Z89321

16-BIT DIGITAL SIGNAL PROCESSOR

FEATURES

- 16-Bit Single Cycle Instructions
- Zero Overhead Hardware Looping
- 16-Bit Data
- Ready Control for Slow Peripherals
- Single Cycle Multiply/Accumulate (100 ns)
- Six-Level Stack
- 512 Words of On-Chip RAM
- Programmable Timer
- 16-Bit I/O Port
- 4K Words of On-Chip Masked ROM
- Three Vectored Interrupts
- Two Conditional Branch Inputs/Two User Outputs
- 24-Bit ALU, Accumulator and Shifter
- IBM® PC Development Tools
- Cost Effective 44-Pin PLCC Package
- CODEC Interface

GENERAL DESCRIPTION

The Z89321 is a second generation, 16-bit, fractional, two's complement CMOS Digital Signal Processor (DSP). Most instructions, including multiply and accumulate, are accomplished in a single clock cycle. The processor contains 1 Kbyte of on-chip data RAM (two blocks of 256 16-bit words), 4K words of program ROM. Also, the processor features a 24-bit ALU, a 16 x 16 multiplier, a 24-bit Accumulator and a shifter. Additionally, the processor contains a six-level stack, three vectored interrupts and two inputs for conditional program jumps. Each RAM block contains a set of three pointers which may be incremented or decremented automatically to affect hardware looping without software overhead. The data RAMs can be simultaneously addressed and loaded to the multiplier for a true single cycle multiply.

The device includes a 16-bit I/O bus for transferring data or for mapping peripherals into the processor address space. Additionally, there are two general purpose user inputs and two user outputs. Operation with slow peripherals is accomplished with a ready input pin.

Development tools for the IBM PC include a relocatable assembler, a linker loader, and an ANSI-C compiler. Also, the development tools include a simulator/debugger, a cross assembler for the TMS320 family assembly code and a hardware emulator.

Notes:
All Signals with a preceding front slash, "/", are active Low, e.g., B//W (WORD is active Low); /B/W (BYTE is active Low, only).

Power connections follow conventional descriptions below:

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

GENERAL DESCRIPTION (Continued)

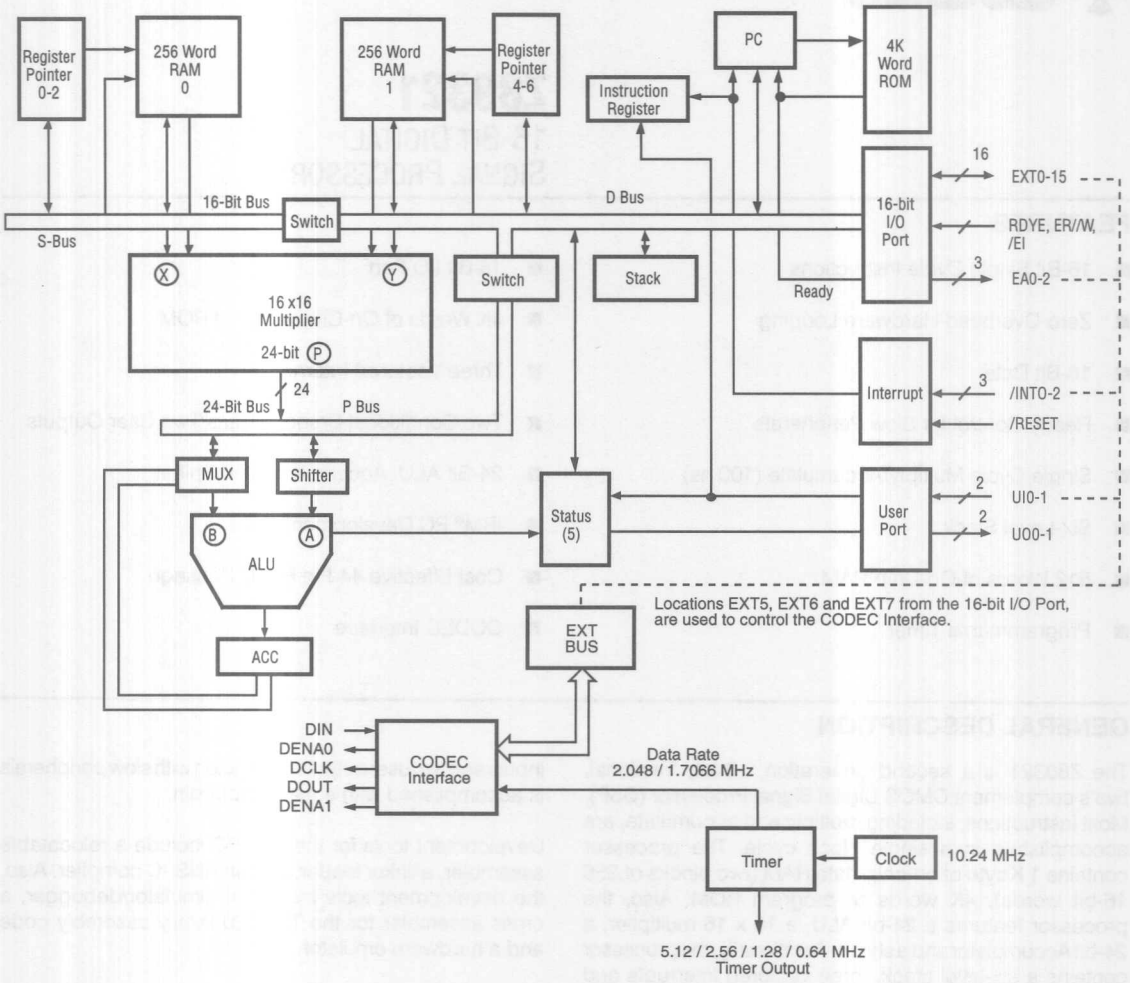


Figure 1. Functional Block Diagram

PIN DESCRIPTION

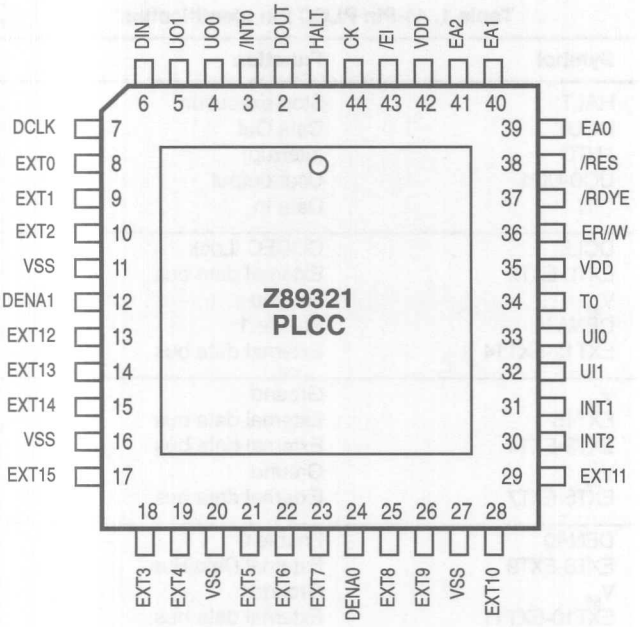


Figure 2. 44-Pin PLCC Pin Assignments

6

PIN DESCRIPTION (Continued)

Table 1. 44-Pin PLCC Pin Identification

No.	Symbol	Function	Direction
1	HALT	Stop execution	Input
2	DOUT	Data Out	Output
3	/INT0	Interrupt	Input
4-5	UO0-UO1	User output	Output
6	DIN	Data In	Input
7	DCLK	CODEC Lock	Output
8-10	EXT0-EXT2	External data bus	Input/Output
11	V_{SS}	Ground	Input
12	DENA1	Enable 1	Output
13-15	EXT12-EXT14	External data bus	Input/Output
16	V_{SS}	Ground	Input
17	EXT15	External data bus	Input/Output
18-19	EXT3-EXT4	External data bus	Input/Output
20	V_{SS}	Ground	Input
21-23	EXT5-EXT7	External data bus	Input/Output
24	DENA0	Enable 0	Output
25-26	EXT8-EXT9	External Data Bus	Input/Output
27	V_{SS}	Ground	Input
28-29	EXT10-EXT11	External data bus	Input/Output
30	/INT2	Interrupt	Input
31	/INT1	Interrupt	Input
32	UI1	User input	Input
33	UI0	User input	Input
34	T0	Timer Output	Output
35	V_{DD}	Power Supply	Input
36	ER/W	R/W for external bus	Output
37	/RDYE	Data ready	Input
38	/RES	Reset	Input
39-41	EA0-EA2	External address bus	Output
42	V_{DD}	Power Supply	Input
43	/EI	Data strobe for external bus	Output
44	CK	Clock	Input

CODEC Interface Controller

External DSP registers ext5 and ext6 are used by External Codec Interface. The accessibility of these devices is driven by the Codec/Timer Control register (ext7).

Two different Codecs can be addressed by the Codec/Timer Control register (ext7). The data can be loaded to Codec0 or Codec1 by writing to ext5 or ext6 correspondingly. In order to receive the data from the Codecs the DSP should read ext5 and ext6.

1. Codec Data Registers - EXT5 and EXT6

The DSP writes data to Codecs using the lower 8 bits of the ext5 and ext6 registers. The 8 remaining upper bits of ext5 and ext6 are reserved, as shown in the following table.

Field	Position	Value	Label
Reserved	fedcba98-------- R W --------76543210 R W	 %NN %NN	 Return "0" No effect Data From Codec Data To Codec

2. Codec/Timer Control Register

The DSP can define the status of the Codecs and the frequency of the Timer output by writing data to a Codec/Timer Control Register.

Field	Position	Value	Label
Reserved	fedcba9876------ R W		 Return "0" No effect
Codec_enable	----------54---- R/W	00 01 10 11	 Disabled C0 enable Reserved Enabled
Div_5/6	------------3--- R/W	0 1	Divided by 6 Divided by 5
Sampling	-------------2-- R/W	0 1	Normal Slow
Timer_rate	--------------10 R/W	00 01 10 11	Div by 2 Div by 4 Div by 8 Div by 16

Codec_enable: This field enables the Codecs. The options are disable both Codecs, enable both Codecs, or enable Codec0 only. Codec1 can not be enabled alone.

Div5/6: This bit defines the speed of codecs. If the bit is set to a "1" the Codec clock frequency is set to 2.048 MHz and the sampling rate equals to 8 KHz. If the bit is reset to "0" - codec clock frequency is equal to 1.7066 MHz, while the sampling rate is set to 6.66 KHz. Upon a POR the bit is reset to "0".

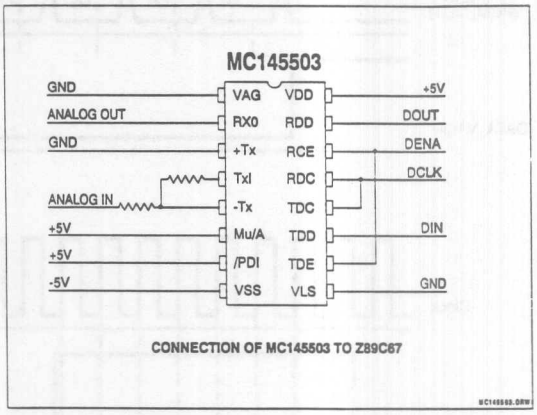
Sampling: This field defines the sampling rate of the Codecs. The sampling rate can be selected from 8 KHz ("0") and 6.66 ("1"). The clock frequency of the Codecs is not controlled by this field. Upon POR the bit is set to a "0".

Timer_rate: This field defines the frequency of the embedded Timer. Upon POR the field is reset to a "00".

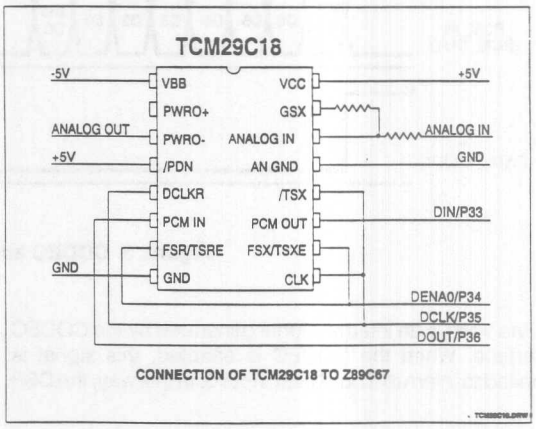
3. The Codec Interface timings.

Codec interface provides the customer with all necessary signals to connect two independent Codec chips. The supported effective data rate for each Codec is 8/6.66 K bytes/sec. The Clock frequency is fixed to 2.048/1.7066 MHz. The following timing diagrams describe the functionality of Codec interface.

Two figures show the connection of Z89321 to popular TI (TCM29C18) and Motorolla's (MC145503) codecs. No additional components are necessary.



CONNECTION OF MC145503 TO Z89C67



CONNECTION OF TCM29C18 TO Z89C67

6

CODEC INTERFACE CONTROLLER (Continued)

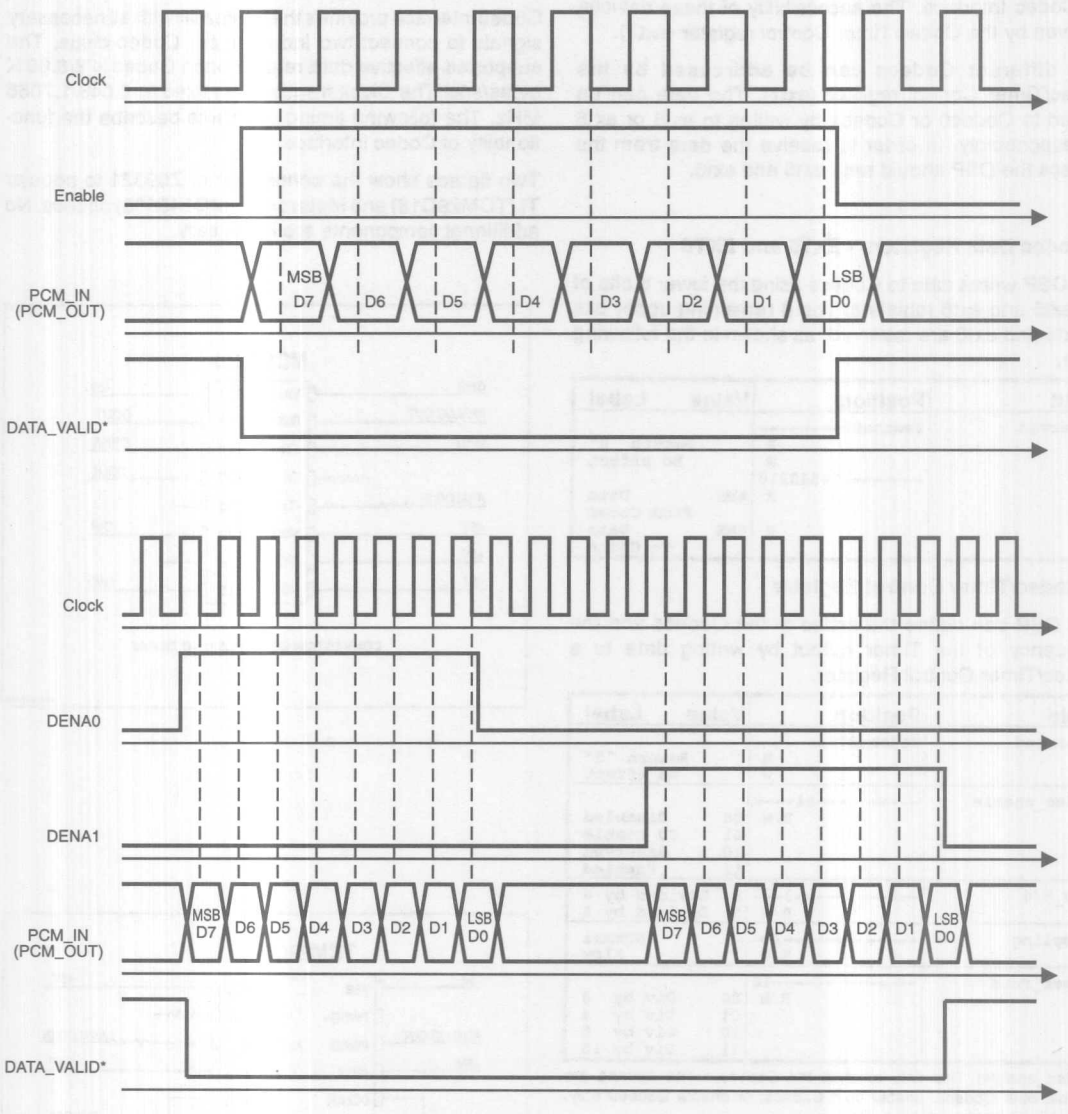


Figure. 3 CODEC Interface Tming Diagram

* Data Valid is an internal signal generated by the CODEC interface. When the CODEC is enabled, this signal is applied to interrupt 0 and user input 0. In this way, the DSP can determine when data is valid either by an interrupt on INT0 or by polling UI0. Under these conditions, INT0 and UI0 are disabled.

PIN FUNCTIONS

CK *Clock* (input). External clock.

EXT15-EXT0 *External Data Bus* (input/output). Data bus for user defined outside registers such as an ADC or DAC. The pins are normally in output mode except when the outside registers are specified as source registers in the instructions. All the control signals exist to allow a read or a write through this bus.

ER/W *External Bus Direction* (output). Data direction signal for EXT-Bus. Data is available from the CPU on EXT15-EXT0 when this signal is Low. EXT-Bus is in input mode (high-impedance) when this signal is High.

EA2-EA0 *External Address* (output). User-defined register address output. One of eight user-defined external registers is selected by the processor with these address pins for read or write operations. Since the addresses are part of the processor memory map, the processor is simply executing internal reads and writes.

/EI *Enable Input* (output). Read/Write timing signal for EXT-Bus. User defined register or the processor can put data on the EXT-Bus during a Low state. Data is read by the external peripheral on the rising edge of /EI. Data is read by the processor on the rising edge of CK not /EI.

HALT *Halt State (input).* Stop Execution Control. The CPU continuously executes NOPs and the program counter remains at the same value when this pin is held High. This signal must be synchronized with CK.

/INT2-/INT0 *Three Interrupts* (input, active Low). Interrupt request 2-0. Interrupts are generated on the rising edge of the input signal. Interrupt vectors for the interrupt service starting address are stored in the program memory locations 0FFFH for /INT0, 0FFEH for /INT1, and 0FFDH for /INT2. Priority is : INT2 = lowest, INT0 = highest.

/RES *Reset* (input, active Low). Asynchronous reset signal. A Low level on this pin generates an internal reset signal. The /RES signal must be kept Low for at least one clock cycle. The CPU pushes the contents of the PC onto the stack and then fetches a new Program Counter (PC) value from program memory address 0FFCH after the reset signal is released.

/RDYE *Data Ready* (input). User-supplied Data Ready signal for data to and from external data bus. This pin stretches the /EI and ER//W lines and maintains data on the address bus and data bus. The ready signal is sampled from the rising edge of the clock with appropriate setup and hold times. The normal write cycle will continue from the next rising clock only if ready is active.

UI1-UI0 *Two Input Pins* (input). General purpose input pins. These input pins are directly tested by the conditional branch instructions. These are asynchronous input signals that have no special clock synchronization requirements.

UO1-UO0 *Two Output Pins* (output). General purpose output pins. These pins reflect the value of two bits in the status register S5 and S6. These bits have no special significance and may be used to output data by writing to the status register.

ADDRESS SPACE

Program Memory. Programs of up to 4K words can be masked into internal ROM. Four locations are dedicated to the vector address for the three interrupts (0FFDH-0FFFH) and the starting address following a Reset (0FFCH). Internal ROM is mapped from 0000H to 0FFFH, and the highest location for program is 0FFBH.

Internal Data RAM. The Z89321 has an internal 512 x 16-bit word data RAM organized as two banks of 256 x 16-bit words each, referred to as RAM0 and RAM1. Each data RAM bank is addressed by three pointers, referred to as Pn:0 (n=0-2) for RAM0 and Pn:1 (n=0-2) for RAM1. The RAM addresses for RAM0 and RAM1 are arranged from 0-255 and 256-511 respectively. The address pointers, which may be written to or read from, are 8-bit registers connected to the lower byte of the internal 16-bit D-Bus and are used to perform no overhead looping. Three addressing modes are available to access the Data RAM: register indirect, direct addressing, and short form direct. These modes are discussed in detail later. The contents of the RAM can be read or written in one machine cycle per word without disturbing any internal registers or status other than the RAM address pointer used for each RAM. The contents of each RAM can be loaded simultaneously into the X and Y inputs of the multiplier.

Registers. The Z89321 has 12 internal registers and up to an additional eight external registers. The external registers are user definable for peripherals such as A/D or D/A or to DMA or other addressing peripherals. External registers are accessed in one machine cycle the same as internal registers.

FUNCTIONAL DESCRIPTION

General. The Z89321 is a high-performance Digital Signal Processor with a modified Harvard-type architecture with separate program and data memory. The design has been optimized for processing power and minimizing silicon space.

Instruction Timing. Many instructions are executed in one machine cycle. Long immediate instructions and Jump or Call instructions are executed in two machine cycles. When the program memory is referenced in internal RAM indirect mode, it takes three machine cycles. In addition, one more machine cycle is required if the PC is selected as the destination of a data transfer instruction. This only happens in the case of a register indirect branch instruction.

An Acc + P => Acc; a(i) * b(j) → P calculation and modification of the RAM pointers, is done in one machine cycle. Both operands, a(i) and b(j), can be located in two independent RAM (0 and 1) addresses.

Multiply/Accumulate. The multiplier can perform a 16-bit x 16-bit multiply or multiply accumulate in one machine cycle using the Accumulator and/or both the X and Y inputs. The multiplier produces a 32-bit result, however, only the 24 most significant bits are saved for the next instruction or accumulation. For operations on very small numbers where the least significant bits are important, the data should first be scaled by eight bits (or the multiplier and multiplicand by four bits each) to avoid truncation errors. Note that all inputs to the multiplier should be fractional two's complement 16-bit binary numbers. This puts them in the range [–1 to 0.9999695], and the result is in 24 bits so that the range is [–1 to 0.9999999]. In addition, if 8000H is loaded into both X and Y registers, the resulting multiplication is considered an illegal operation as an overflow would result. Positive one cannot be represented in fractional notation, and the multiplier will actually yield the result 8000H x 8000H = 8000H (–1 x –1 = –1).

ALU. The 24-bit ALU has two input ports, one of which is connected to the output of the 24-bit Accumulator. The other input is connected to the 24-bit P-Bus, the upper 16 bits of which are connected to the 16-bit D-Bus. A shifter between the P-Bus and the ALU input port can shift the data by three bits right, one bit right, one bit left or no shift.

Hardware Stack. A six-level hardware stack is connected to the D-Bus to hold subroutine return addresses or data. The Call instruction pushes PC+2 onto the stack. The RET instruction pops the contents of the stack to the PC.

User Inputs. The Z89321 has two inputs, UI0 and UI1, which may be used by Jump and Call instructions. The Jump or Call tests one of these pins and if appropriate, jumps to a new location. Otherwise, the instruction behaves like a NOP. These inputs are also connected to the status register bits S10 and S11 which may be read by the appropriate instruction (Figure 4).

User Outputs. The status register bits S5 and S6 connect directly to UO0 and UO1 pins and may be written to by the appropriate instruction.

Interrupts. The Z89321 has three positive edge-triggered interrupt inputs. An interrupt is acknowledged at the end of any instruction execution. It takes two machine cycles to enter an interrupt instruction sequence. The PC is pushed onto the stack. A RET instruction transfers the contents of the stack to the PC and decrements the stack pointer by one word. The priority of the interrupts is INT0 = highest, INT2 = lowest.

Registers. The Z89321 has 12 physical internal registers and up to eight user-defined external registers. The EA2-EA0 determines the address of the external registers. The /EI, /RDYE, and ER/W signals are used to read or write from the external registers.

REGISTERS

There are 12 internal registers which are defined below:

Register	Register Definition
P	Output of Multiplier, 24-Bit
X	X Multiplier Input, 16-Bit
Y	Y Multiplier Input, 16-Bit
A	Accumulator, 24-Bit
SR	Status Register, 16-Bit
Pn:b	Six Ram Address Pointers, 8-Bit Each
PC	Program Counter, 16-Bit

The following are virtual registers as physical RAM does not exist on the chip.

EXTn	External Registers, 16-bit
BUS	D-Bus
Dn:b	Eight Data Pointers

P holds the result of multiplications and is read-only.

X and **Y** are two 16-bit input registers for the multiplier. These registers can be utilized as temporary registers when the multiplier is not being used.

A is a 24-bit Accumulator. The output of the ALU is sent to this register. When 16-bit data is transferred into this register, it goes into the 16 MSB's and the least significant eight bits are set to zero. Only the upper 16 bits are transferred to the destination register when the Accumulator is selected as a source register in transfer instructions.

Pn:b are the pointer registers for accessing data RAM. (n=0,1,2 refer to the pointer number) (b=0,1 refers to RAM Bank 0 or 1). They can be directly read from or written to, and can point to locations in data RAM or Program Memory.

EXTn are external registers (n=0 to 7). There are eight 16-bit registers here for mapping external devices into the address space of the processor. Note that the actual register RAM does not exist on the chip, but would exist as part of the external device such as an ADC result latch.

BUS is a read-only register which, when accessed, returns the contents of the D-Bus.

Dn:b refer to possible locations in RAM that can be used as a pointer to locations in program memory. The programmer decides which location to choose from two bits in the status register and two bits in the operand. Thus, only the lower 16 possible locations in RAM can be specified. At any one time there are eight usable pointers, four per bank, and the four pointers are in consecutive locations in RAM. For example, if S3/S4 = 01 in the status register, then D0:0/D1:0/D2:0/D3:0 refer to locations 4/5/6/7 in RAM Bank 0. Note that when the data pointers are being written to, a number is actually being loaded to Data RAM, so they can be used as a limited method for writing to RAM.

REGISTERS (Continued)

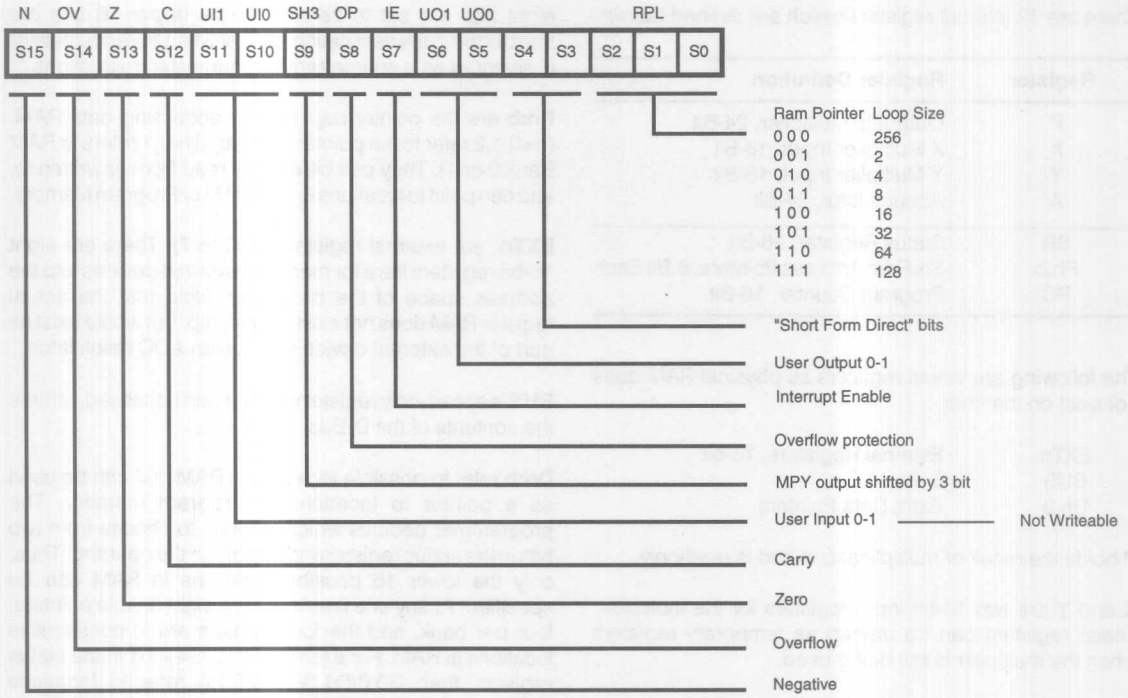


Figure 4. Status Register

SR is the status register (Figure 4) which contains the ALU status and certain control bits as shown in the following table.

Table 2. Status Register Bit Functions

Status Register Bit	Function
S15 (N)	ALU Negative
S14 (OV)	ALU Overflow
S13 (Z)	ALU Zero
S12 (L)	Carry
S11 (UI1)	User Input 1
S10 (UI0)	User Input 0
S9 (SH3)	MPY Output Shifted by 3 Bits
S8 (OP)	Overflow Protection
S7 (IE)	Interrupt Enable
S6 (UO1)	User Output 1
S5 (UO0)	User Output 0
S4-S3	"Short Form Direct" Bits
S2-S0 (RPL)	RAM Pointer Loop Size

Table 3. RPL Description

S2	S1	S0	Loop Size
0	0	0	256
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The status register may always be read in its entirety. S15-S10 are set/reset by the hardware and can only be read by software. S9-S0 can be written by software.

S15-S12 are set/reset by the ALU after an operation. S11-S10 are set/reset by the user inputs. S6-S0 are control bits described elsewhere. S7 enables interrupts. S8, if 0 (reset), allows the hardware to overflow. If S8 is set, the hardware clamps at maximum positive or negative values instead of overflowing. If S9 is 0, the shifter shifts data one bit left or right. If S9 is set and a shift is called for on a multiply instruction, then the shifter shifts the result three bits right instead of one bit right.

PC is the Program Counter. When this register is assigned as a destination register, one NOP machine cycle is added automatically to adjust the pipeline timing.

RAM ADDRESSING

The address of the RAM is specified in one of three ways (Figure 5):

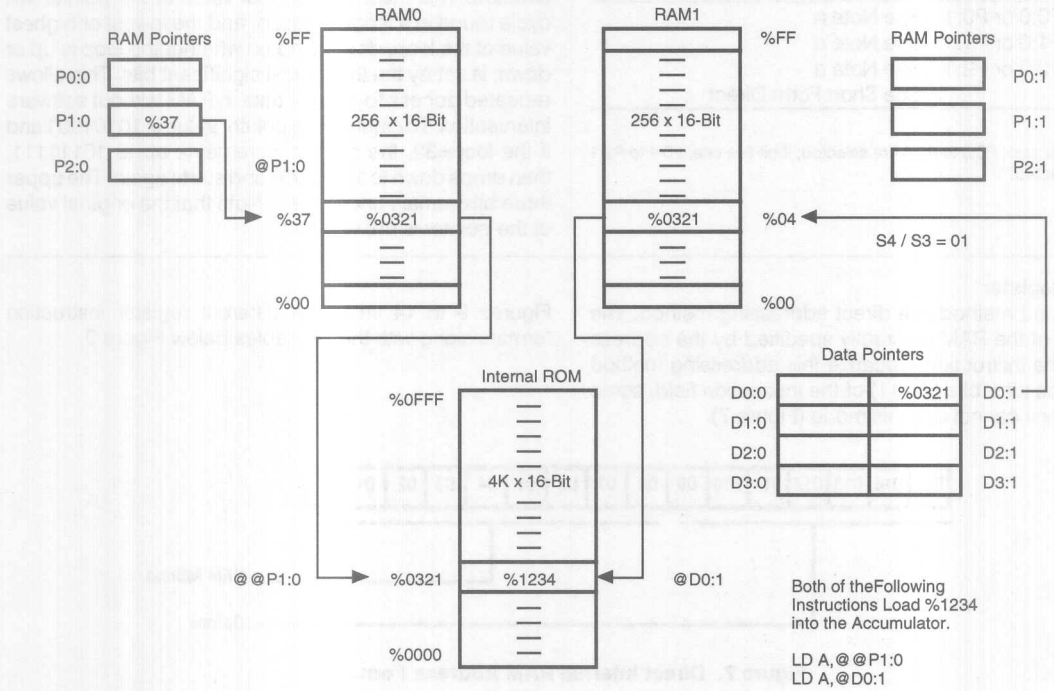


Figure 5. RAM, ROM, and Pointer Architecture

Register Indirect

Pn:b n=0-2, b=0-1

The most commonly used method is a register indirect addressing method, where the RAM address is specified by one of the three RAM address pointers (n) for each bank (b). Each source/destination field in Figures 6 and 9 may be used by an indirect instruction to specify a register pointer and its modification after execution of the instruction.

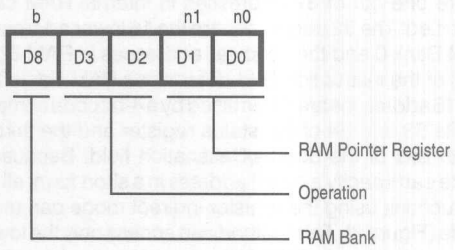


Figure 6. Indirect Register

RAM ADDRESSING (Continued)

The register pointer is specified by the first and second bits in the source/destination field and the modification is specified by the third and fourth bits according to the following table:

D3-D0		Meaning
00xx	NOP	No Operation
01xx	+1	Simple Increment
10xx	–1/LOOP	Decrement Modulo the Loop Count
11xx	+1/LOOP	Increment Modulo the Loop Count
xx00	P0:0 or P0:1	See Note a
xx01	P1:0 or P1:1	See Note a
xx10	P2:0 or P2:1	See Note a
xx11		See Short Form Direct

Notes:
a. If bit 8 is zero, P0:0 to P2:0 are selected; if bit 8 is one, P0:1 to P2:1 are selected.

When LOOP mode is selected, the pointer to which the loop is referring will cycle up or down, depending on whether a –LOOP or +LOOP is specified. The size of the loop is obtained from the least significant three bits of the Status Register. The increment or decrement of the register is accomplished modulo the loop size. As an example, if the loop size is specified as 32 by entering the value 101 into bits 2-0 of the Status Register (S2-S0) and an increment +LOOP is specified in the address field of the instruction, i.e., the RPi field is 11xx, then the register specified by RPi will increment, but only the least significant five bits will be affected. This means the actual value of the pointer will cycle round in a length 32 loop, and the lowest or highest value of the loop, depending on whether the loop is up or down, is set by the three most significant bits. This allows repeated access to a set of data in RAM without software intervention. To clarify, if the pointer value is 10101001 and if the loop=32, the pointer increments up to 10111111, then drops down to 10100000 and starts again. The upper three bits remain unchanged. Note that the original value of the pointer is not retained.

Direct Register

The second method is a direct addressing method. The address of the RAM is directly specified by the address field of the instruction. Because this addressing method consumes nine bits (0-511) of the instruction field, some instructions cannot use this mode (Figure 7).

Figures 9 to 14 show the different register instruction formats along with the two tables below Figure 9.

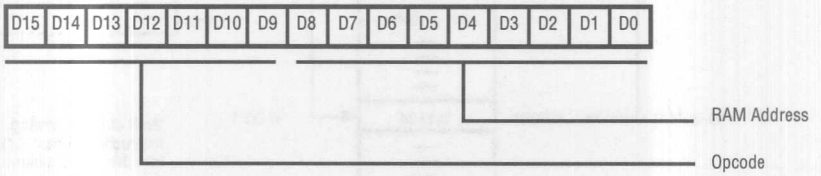


Figure 7. Direct Internal RAM Address Format

Short Form Direct

Dn:b n = 0-3, b = 0-1

The last method is called Short Form Direct Addressing, where one out of 32 addresses in internal RAM can be specified. The 32 addresses are the 16 lower addresses in RAM Bank 0 and the 16 lower addresses in RAM Bank 1. Bit 8 of the instruction field determines RAM Bank 0 or 1. The 16 addresses are determined by a 4-bit code comprised of bits S3 and S4 of the status register and the third and fourth bits of the Source/Destination field. Because this mode can specify a direct address in a short form, all of the instructions using the register indirect mode can use this mode (Figure 8). This method can access only the lower 16 addresses in the both RAM banks and as such has limited use. The main purpose is to specify a data register, located in the RAM bank, which can then be used to point to a program memory location. This facilitates down-loading look-up tables etc. from program memory to RAM.

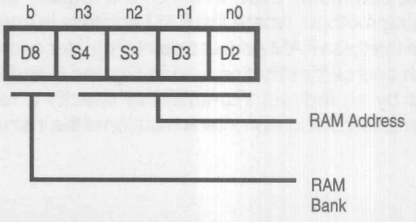
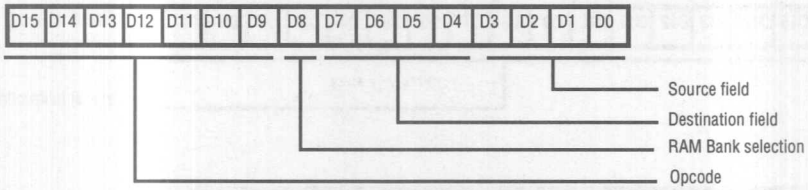


Figure 8. Short Form Direct Address

INSTRUCTION FORMAT



Note:
Source/Destination fields can specify either register or RAM addresses in RAM pointer indirect mode.

Figure 8. General Instruction Format

Table 4. Registers

Source/Destination	Register
0000	BUS**
0001	X
0010	Y
0011	A
0100	SR
0101	STACK
0110	PC
0111	P**
1000	EXT0
1001	EXT1
1010	EXT2
1011	EXT3
1100	EXT4
1101	EXT5
1110	EXT6
1111	EXT7

Table 5. Register Pointers Field

Source/Destination	Meaning
00xx	NOP
01xx	+1
10xx	–1/LOOP
11xx	+1/LOOP
xx00	P0:0 or P0:1*
xx01	P1:0 or P1:1*
xx10	P2:0 or P2:1*
xx11	Short Form Direct Mode

Notes:
* If RAM Bank bit is 0, then Pn:0 are selected.
If RAM Bank bit is 1, then Pn:1 are selected.
** Read only.
*** When the short form direct mode is selected, 00000-01111 or 10000-11111 are used as RAM addresses.

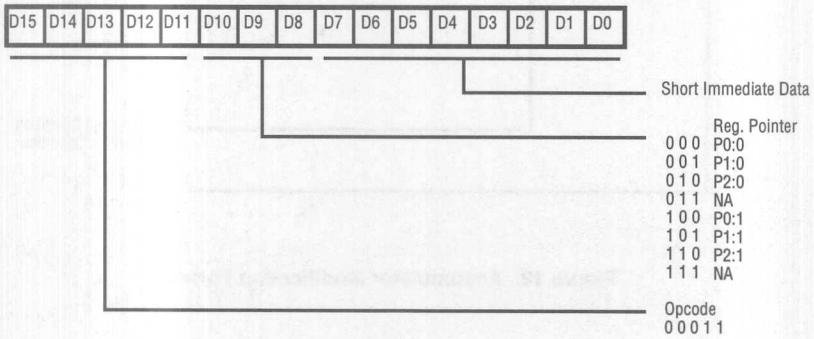


Figure 10. Short Immediate Data Load Format

6

INSTRUCTION FORMAT (Continued)

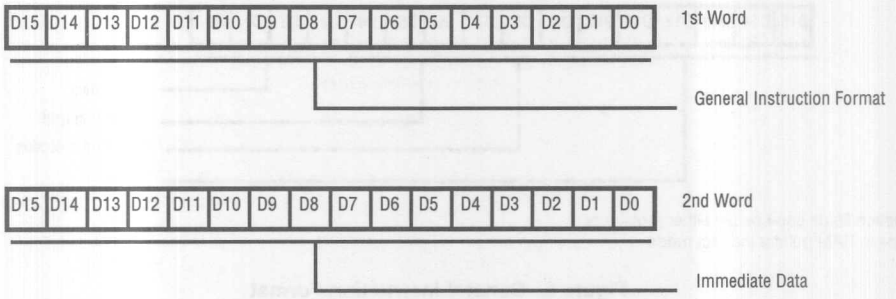


Figure 11. Immediate Data Load Format

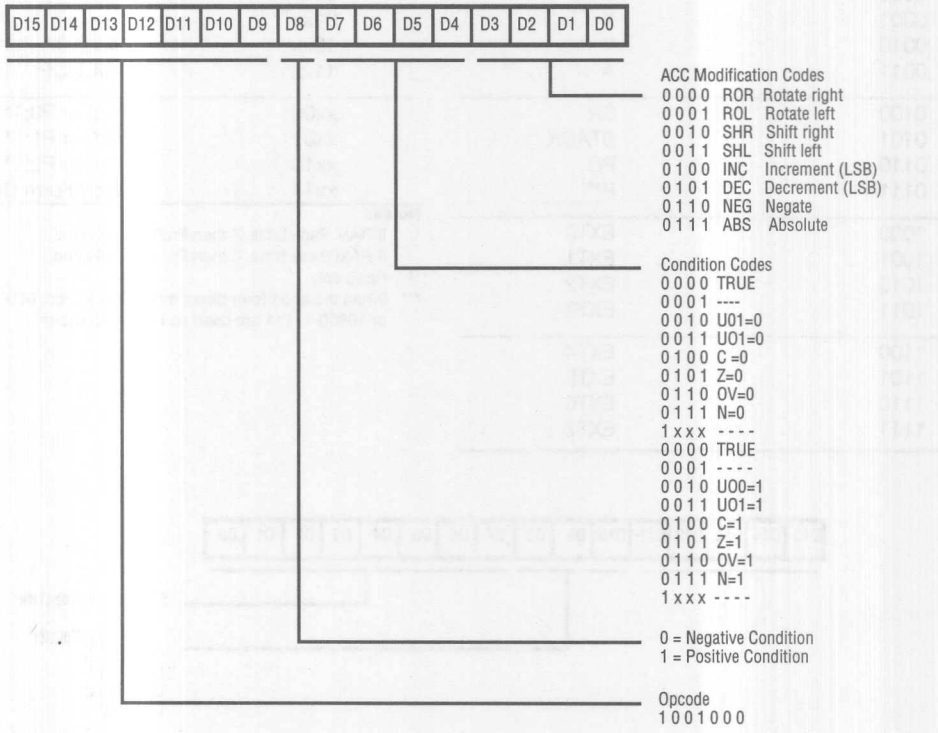


Figure 12. Accumulator Modification Format

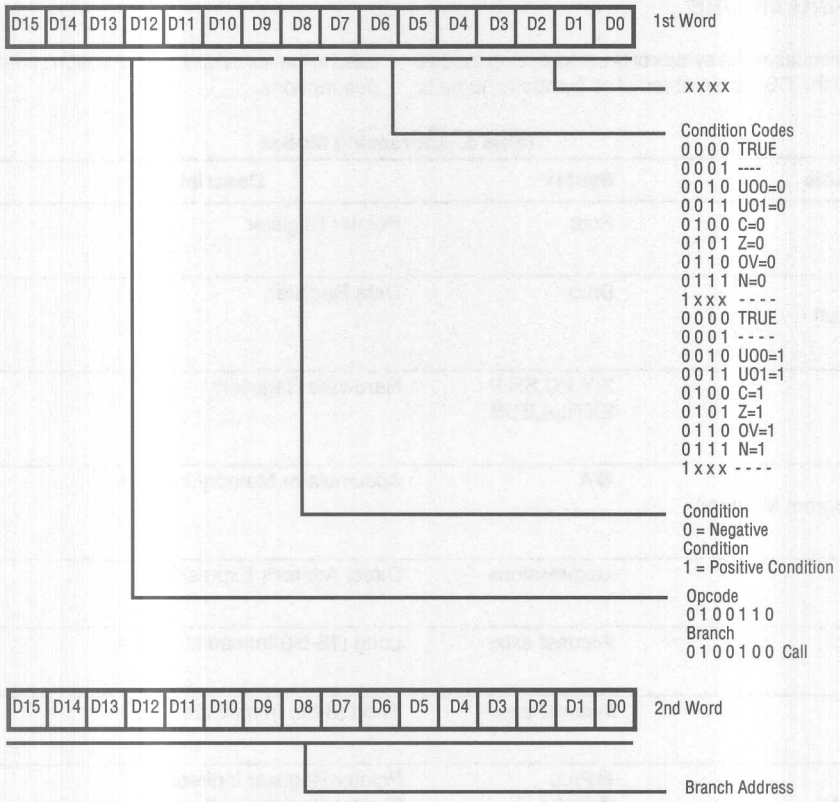


Figure 13. Branching Format

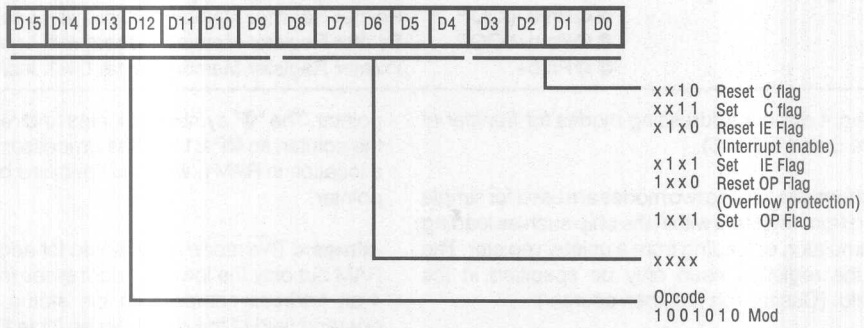


Figure 14. Flag Modification Format

6

ADDRESSING MODES

This section discusses the syntax of the addressing modes supported by the DSP assembler. The symbolic name is used in the discussion of instruction syntax in the instruction descriptions.

Table 5. Addressing Modes

Symbolic Name	Syntax	Description
<pregs>	Pn:b	Pointer Register
<dregs> (Points to RAM)	Dn:b	Data Register
<hwregs>	X,Y,PC,SR,P EXTn,A,BUS	Hardware Registers
<accind> (Points to Program Memory)	@A	Accumulator Memory Indirect
<direct>	<expression>	Direct Address Expression
<limm>	#<const exp>	Long (16-bit) Immediate Value
<simm>	#<const exp>	Short (8-bit) Immediate Value
<regind> (Points to RAM)	@Pn:b @Pn:b+ @Pn:b–LOOP @Pn:b+LOOP	Pointer Register Indirect Pointer Register Indirect with Increment Pointer Register Indirect with Loop Decrement Pointer register Indirect with Loop Increment
<memind> (Points to Program Memory)	@@Pn:b @Dn:b @@Pn:b–LOOP @@Pn:b+LOOP @@Pn:b+	Pointer Register Memory Indirect Data Register Memory Indirect Pointer Register Memory Indirect with Loop Decrement Pointer Register Memory Indirect with Loop Increment Pointer Register Memory Indirect with Increment

There are eight distinct addressing modes for transfer of data (Figure 5 and Table 5).

<pregs>, <hwregs>. These two modes are used for simple loads to and from registers within the chip such as loading to the Accumulator, or loading from a pointer register. The names of the registers need only be specified in the operand field. (Destination first then source.)

<regind>. This mode is used for indirect accesses to the data RAM. The address of the RAM location is stored in the pointer. The "@" symbol indicates "indirect" and precedes the pointer, so @P1:1 tells the processor to read or write to a location in RAM1, which is specified by the value in the pointer.

<dregs>. This mode is also used for accesses to the data RAM but only the lower 16 addresses in either bank. The 4-bit address comes from the status register and the operand field of the data pointer. Note that data registers are typically used not for addressing RAM, but loading data from program memory space.

<memind>. This mode is used for indirect, indirect accesses to the program memory. The address of the memory is located in a RAM location, which is specified by the value in a pointer. So @@P1:1 tells the processor to read (write is not possible) from a location in memory, which is specified by a value in RAM, and the location of the RAM is in turn specified by the value in the pointer. Note that the data pointer can also be used for a memory access in this manner, but only one "@" precedes the pointer. In both cases the memory address stored in RAM is incremented by one each time the addressing mode is used to allow easy transfer of sequential data from program memory.

<accind>. Similar to the previous mode, the address for the program memory read is stored in the Accumulator. @A in the second operand field loads the number in memory specified by the address in A.

<direct>. The direct mode allows read or write to data RAM from the Accumulator by specifying the absolute address of the RAM in the operand of the instruction. A number between 0 and 255 indicates a location in RAM0, and a number between 256 and 511 indicates a location in RAM1.

<limm>. This indicates a long immediate load. A 16-bit word can be copied directly from the operand into the specified register or memory.

<simm>. This can only be used for immediate transfer of 8-bit data in the operand to the specified RAM pointer.

CONDITION CODES

The following defines the condition codes supported by the DSP assembler. If the instruction description refers to the <cc> (condition code) symbol in one of its addressing modes, the instruction will only execute if the condition is true.

Name	Description
C	Carry
EQ	Equal (same as Z)
F	False
IE	Interrupts Enabled
MI	Minus
NC	No Carry
NE	Not Equal (same as NZ)
NIE	Not Interrupts Enabled
NOV	Not Overflow
NU0	Not User Zero

Name	Description
NU1	Not User One
NZ	Not zero
OV	Overflow
PL	Plus (Positive)
U0	User Zero
U1	User One
UGE	Unsigned Greater Than or Equal (Same as NC)
ULT	Unsigned Less Than (Same as C)
Z	Zero

INSTRUCTION DESCRIPTIONS

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
ABS	Absolute Value	ABS[<cc>,]<src>	<cc>,A	1	1	ABS NC,A
			A	1	1	ABS A
ADD	Addition	ADD<dest>,<src>	A,<pregs>	1	1	ADD A,#128
			A,<dregs>	1	1	ADD A,D0:1
			A,<limm>	2	2	ADD A,@@LOOP
			A,<memind>	1	3	ADD A,@P2:1+
			A,<direct>	1	1	ADD A,X
			A,<regind>	1	1	
			A,<hwregs>	1	1	
AND	Bitwise AND	AND<dest>,<src>	A,<pregs>	1	1	AND A,#128
			A,<dregs>	1	1	AND A,D0:1
			A,<limm>	2	2	AND A,@@P0:0+LOOP
			A,<memind>	1	3	AND A,@P2:1+
			A,<direct>	1	1	
			A,<regind>	1	1	AND A,X
			A,<hwregs>	1	1	
CALL	Subroutine call	CALL [<cc>,]<address>	<cc>,<direct>	2	2	CALL sub1
			<direct>	2	2	CALL Z,sub2
CCF	Clear carry flag	CCF	None	1	1	CCF
CIEF	Clear Carry Flag	CIEF	None	1	1	CIEF
COPF	Clear OP flag	COPF	None	1	1	COPF
CP	Comparison	CP<src1>,<src2>	A,<pregs>	1	1	CP A,P0:0
			A,<dregs>	1	1	CP A,D3:1
			A,<memind>	1	3	CP A,#512
			A,<direct>	1	1	CP A,@@P0:1
			A,<regind>	1	1	CP A,LABEL
			A,<hwregs>	1	1	CP A,@D0:0
						CP A,X
DEC	Decrement	DEC [<cc>,]<dest>	<cc>A,	1	1	DEC NZ,A
			A	1	1	DEC A
INC	Increment	INC [<cc>,] <dest>	<cc>,A	1	1	INC NZ,A
			A	1	1	INC A
JP	Jump	JP [<cc>,]<address>	<cc>,<direct>	2	2	JP NIE,Label
			<direct>	2	2	JP Label

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
LD	Load destination with source	LD<dest>,<src>	A,<hwregs>	1	1	LD A,X
			A,<dregs>	1	1	LD A,D0:0
			A,<pregs>	1	1	LD A,P0:1
			A,<regind>	1	1	LD A,@@P1:1
			A,<memind>	1	3	LD A,MEMADDR
			A,<direct>	1	1	LD MEMADDR,A
			<direct>,A	1	1	LD D0:1,A
			<dregs>,<hwregs>	1	1	LD P1:0#128
			<pregs>,<simm>	1	1	LDP1:1,X
			<pregs>,<hwregs>	1	1	LD@P0:0+LOOP,#1234
			<regind>,<limm>	1	1	LD @P1:1+,X
			<regind>,<hwregs>	1	1	LD X,P0:0
			<hwregs>,<pregs>	1	1	LD Y,P0:0
			<hwregs>,<dregs>	1	1	LD SR,#%1023
			<hwregs>,<limm>	2	2	LD PC,(A)
			<hwregs>,<accind>	1	3	LD X,@@P0:0
			<hwregs>,<memind>	1	3	LD Y,@P1:0–LOOP
			<hwregs>,<regind>	1	1	LD SR,X
			<hwregs>,<hwregs>	1	1	
			Note: When <dest> is <hwregs>, <dest> cannot be P. Note: When <dest> is <hwregs> and <src> is <hwregs>, <dest> cannot be EXTn if <src> is EXTn, <dest> cannot be X if <src> is X, <dest> cannot be SR if <src> is SR. Note: When <src> is <accind> <dest> cannot be A.			
MLD	Multiply	MLD<srcl>,<srcl>[,<bank switch>]	<hwregs>,<regind>	1	1	MLD A@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MLD A@P1:0,OFF
			<regind>,<regind>	1	1	MLD @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MLD @P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <band switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			
MPYA	Multiply and add	MPYA <srcl>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYA A@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYA A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYA @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYA@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register. Note: <hwregs> for src1 cannot be X. Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, the <bank switch> defaults to ON.			

INSTRUCTION DESCRIPTIONS (Continued)

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
MPYS	Multiply and subtract	MPYS<src1>,<src2>[,<bank switch>]	<hwregs>,<regind>	1	1	MPYS A,@P0:0
			<hwregs>,<regind>,<bank switch>	1	1	MPYS A,@P1:0,OFF
			<regind>,<regind>	1	1	MPYS @P1:1,@P2:0
			<regind>,<regind>,<bank switch>	1	1	MPYS@P0:1,@P1:0,ON
			Note: If src1 is <regind> it must be a bank 1 register. Src2's <regind> must be a bank 0 register.			
			Note: <hwregs> for src1 cannot be X.			
			Note: For the operands <hwregs>, <regind> the <bank switch> defaults to OFF. For the operands <regind>, <regind> the <bank switch> defaults to ON.			
NEG	Negate	NEG <cc>,A	<cc>, A	1	1	NEG NZ,A
			A	1	1	NEG A
NOP	No operation	NOP	None	1	1	NOP
OR	Bitwise OR	OR <dest>,<src>	A, <pregs>	1	1	OR A,#128
			A, <dregs>	1	1	OR A, D0:1
			A, <limm>	2	2	OR A,@@P0:0+LOOP
			A, <memind>	1	3	OR A,@P2:1+
			A, <direct>	1	1	OR A, X
			A, <regind>	1	1	
			A, <hwregs>	1	1	
POP	Pop value from stack	POP <dest>	<pregs>	1	1	POP P0:0
			<pregs>	1	1	POP D0:1
			<regind>	1	1	POP @P0:0
			<hwregs>	1	1	POP A
						POP BUS
PUSH	Push value onto stack	PUSH <src>	<pregs>	1	1	PUSH P0:0
			<dregs>	1	1	PUSH D0:1
			<regind>	1	1	PUSH @P0:0
			<hwregs>	1	1	PUSH A
			<limm>	2	2	PUSH BUS
			<accind>	1	3	PUSH #12345
			<memind>	1	3	PUSH @A
						PUSH @@P0:0
RET	Return from subroutine	RET	None	1	2	RET
RL	Rotate Left	RL <cc>,A	<cc>,A	1	1	RL NZ,A
			A	1	1	RL A
RR	Rotate Right	RR <cc>,A	<cc>,A	1	1	RR NZ,A
			A	1	1	RR A

Inst.	Description	Synopsis	Operands	Words	Cycles	Examples
SCF	Set C flag	SCF	None	1	1	SCF
SIEF	Set IE flag	SIEF	None	1	1	SIEF
SLL	Shift left logical	SLL	[<cc>,]A	1	1	SLL NZ,A
			A	1	1	SLL A
SOPF	Set OP flag	SOPF	None	1	1	SOPF
SRA	Shift right arithmetic	SRA<cc>,A	<cc>,A	1	1	SRA NZ,A
			A	1	1	SRA A
SUB	Subtract	SUB<dest>,<src>	A,<pregs>	1	1	SUB A,#128
			A,<dregs>	1	1	SUB A,D0:1
			A,<limm>	2	2	SUB A,@@P0:0+LOOP
			A, <memind>	1	3	SUB A,@P2:1+
			A, <direct>	1	1	SUB A,X
			A, <regind>	1	1	
			A, <hwregs>	1	1	
XOR	Bitwise exclusive OR	XOR <dest>,<src>	A, <pregs>	1	1	XOR A,#128
			A, <dregs>	1	1	XOR A,D0:1
			A, <limm>	2	2	XOR A,@@P0:0+LOOP
			A, <memind>	1	3	XOR A,@P2:1+
			A, <direct>	1	1	XOR A, X
			A, <regind>	1	1	
			A, <hwregs>	1	1	

Bank Switch Enumerations. The third (optional) operand of the MLD, MPYA and MPYS instructions represents whether a bank switch is set on or off. To more clearly represent this two keywords are used (ON and OFF) which state the direction of the switch. These keywords are referred to in the instruction descriptions via the <bank switch> symbol.

ABSOLUTE MAXIMUM RATINGS

Symbol	Description	Min	Max	Units
V_{CC}	Supply Voltage (*)	-0.3	+7.0	V
T_{STG}	Storage Temp	-65°	+150°	C
T_A	Oper Ambient Temp		†	C

Notes:
* Voltage on all pins with respect to GND.
† See Ordering Information.

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended period may affect device reliability.

STANDARD TEST CONDITIONS

The characteristics listed below apply for standard test conditions as noted. All voltages are referenced to ground. Positive current flows into the referenced pin (Figure 15.)

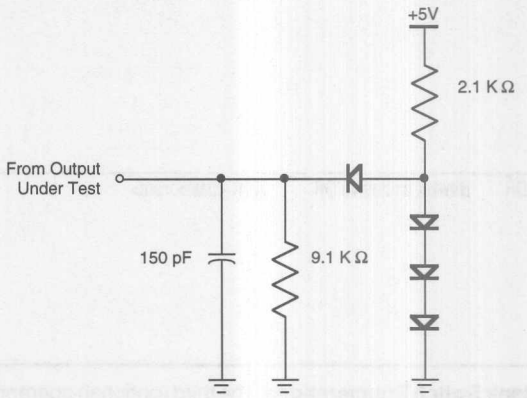


Figure 15. Test Load Diagram

DC ELECTRICAL CHARACTERISTICS

(V_{DD}= 5V ±10%, T_A = –40°C to +105°C unless otherwise specified)

Symbol	Parameter	Condition	Min.	Max.	Units
I_{DD}	Supply Current	V_{DD} = 5.25V fclock = 10 MHz		60	mA
I_{DC}	DC Power Consumption	V_{DD} = 5.25V	1	5	mA
V_{IH}	Input High Level		0.9 V_{DD}		V
V_{IL}	Input Low Level			0.1 V_{DD}	V
IL	Input Leakage			1	µA
V_{OH}	Output High Voltage	I_{OH} = –100 µA	V_{DD}–0.2		V
V_{OL}	Output Low Voltage	I_{OL} = 0.5 mA		0.5	V
I_{FL}	Output Floating Leakage Current			5	µA

AC ELECTRICAL CHARACTERISTICS

(V_{DD} = 5V ±10%, T_A = –40°C to +105°C unless otherwise specified)

Symbol	Parameter	Min.	Max.	Units
TCY	Clock Cycle Time	100	1000	ns
PWW	Clock Pulse Width	45		ns
Tr	Clock Rise Time*	2	4	ns
Tf	Clock Fall Time*	2	4	ns
TEAD	EA,ER//W Delay from CK	15	25	ns
TXVD	EXT Data Output Valid from CK	5	25	ns
TXWH	EXT Data Output Hold from CK	15		ns
TXRS	EXT Data Input Setup Time	15		ns
TXRH	EXT Data Input Hold from CK	0	15	ns
TIED	/EI Delay Time from CK	0	5	ns
RDYS	Ready Setup Time	10		ns
RDYH	Ready Hold Time	0		ns

6

AC TIMING DIAGRAM

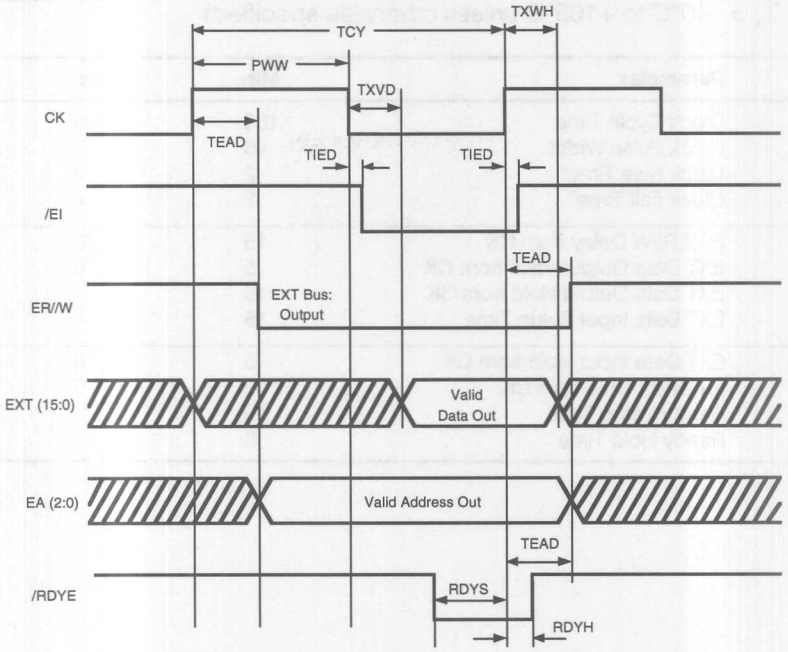


Figure 16. WRITE to external device timing

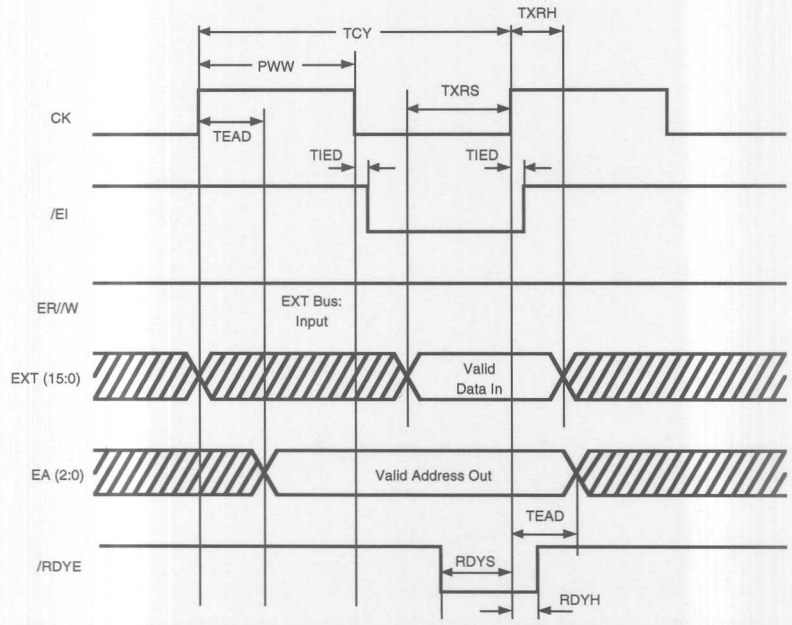
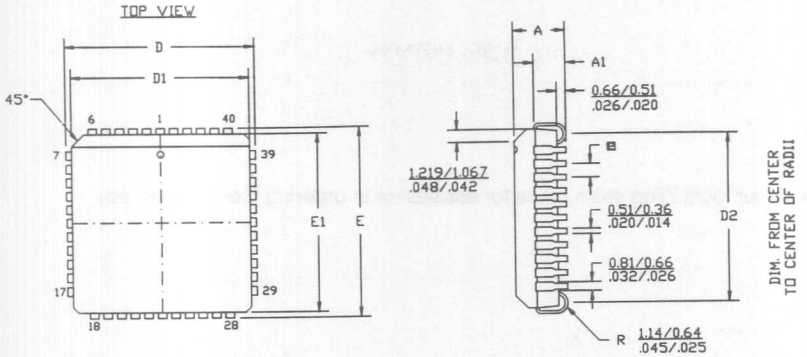


Figure 17b. READ from external device timing

PACKAGE INFORMATION



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.27	4.57	.168	.180
A1	2.67	2.92	.105	.115
D/E	17.40	17.65	.685	.695
D1/E1	16.51	16.66	.650	.656
D2	15.24	16.00	.600	.630
B	1.27 TYP		.050 TYP	

NOTES:
1. CONTROLLING DIMENSIONS : INCH
2. LEADS ARE COPLANAR WITHIN .004 IN.
3. DIMENSION : MM / INCH

44-Pin PLCC Package Diagram

6

ORDERING INFORMATION

Z89321

10 MHz
44-pin PLCC
Z8932110VSC
Z8932110VEC

For fast results, contact your local Zilog sales office for assistance in ordering the part desired.

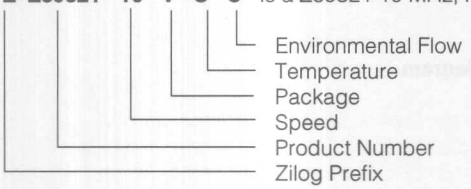
Package
V = Plastic PLCC

Temperature
S = 0°C to +70°C
E = -40°C to +105°C

Speed
10 = 10 MHz

Environmental
C = Plastic Standard

Example:
Z Z89321 10 V S C is a Z89321 10 MHz, PLCC, 0°C to +70°C, Plastic Standard Flow.



Z8®/SUPER8™ MICROCONTROLLER FAMILY

Databooks	Part No	Unit Cost
Z8 Microcontrollers Databook (includes the following documents)	**DC-8275-04**	**5.00**

Z8 CMOS Microcontrollers
Z86C00/C10/C20 MCU OTP Product Specification
Z86C06 Z8 CCP™ Preliminary Product Specification
Z86C08 8-Bit MCU Product Specification
Z86E08 Z8 OTP MCU Product Specification
Z86C09/19 Z8 CCP Product Specification
Z86E19 Z8 OTP MCU Advance Information Specification
Z86C11 Z8 MCU Product Specification
Z86C12 Z8 ICE Product Specification
Z86C21 Z8 MCU Product Specification
Z86E21/Z86E22 OTP Product Specification
Z86C30 Z8 CCP Product Specification
Z86E30 Z8 OTP CCP Product Specification
Z86C40 Z8 CCP Product Specification
Z86E40 Z8 OTP CCP Product Specification
Z86C27/97 Z8 DTC™ Product Specification
Z86127 Low-Cost Digital Television Controller Adv. Info. Spec.
Z86C50 Z8 CCP ICE Advance Information Specification
Z86C61 Z8 MCU Advance Information Specification
Z86C62 Z8 MCU Advance Information Specification
Z86C89/C90 CMOS Z8 CCP Product Specification
Z86C91 Z8 ROMless MCU Product Specification
Z86C93 Z8 ROMless MCU Preliminary Product Specification
Z86C94 Z8 ROMless MCU Product Specification
Z86C96 Z8 ROMless MCU Advance Information Specification
Z88C00 CMOS Super8 MCU Advance Information Specification

Z8 NMOS Microcontrollers
Z8600 Z8 MCU Product Specification
Z8601/03/11/13 Z8 MCU Product Specification
Z8602 8-Bit Keyboard Controller Preliminary Product Spec.
Z8604 8-Bit MCU Product Specification
Z8612 Z8 ICE Product Specification
Z8671 Z8 MCU With BASIC/Debug Interpreter Product Spec.
Z8681/82 Z8 MCU ROMless Product Specification
Z8691 Z8 MCU ROMless Product Specification
Z8800/01/20/22 Super8 ROMless/ROM Product Specification

Peripheral Products
Z86128 Closed-Captioned Controller Adv. Info. Specification
Z765A Floppy Disk Controller Product Specification
Z5380 SCSI Product Specification
Z53C80 SCSI Advance Information Specification

Z8 Application Notes and Technical Articles
Zilog Family On-Chip Oscillator Design
Z86E21 Z8 Low Cost Thermal Printer
Z8 Applications for I/O Port Expansions
Z86C09/19 Low Cost Z8 MCU Emulator
Z8602 Controls A 101/102 PC/Keyboard
The Z8 MCU Dual Analog Comparator
The Z8 MCU In Telephone Answering Systems
Z8 Subroutine Library
A Comparison of MCU Units
Z86xx Interrupt Request Registers
Z8 Family Framing
A Programmer's Guide to the Z8 MCU
Memory Space and Register Organization

Super8 Application Notes and Technical Articles
Getting Started with the Zilog Super8
Polled Async Serial Operations with the Super8
Using the Super8 Interrupt Driven Communications
Using the Super8 Serial Port with DMA
Generating Sine Waves with Super8
Generating DTMF Tones with Super8
A Simple Serial Parallel Converter Using the Super8

Additional Information
Z8 Support Products
Zilog Quality and Reliability Report
Literature List
Package Information
Ordering Information

Z8®/SUPER8™ MICROCONTROLLER FAMILY (Continued)

Databooks	Part No	Unit Cost
Digital Signal Processor Databook (includes the following documents) Z89C00 16-Bit Digital Signal Processor Preliminary Product Specification Z89C00 DSP Application Note "Understanding Q15 Two's Complement Fractional Multiplication" Z89120, Z89920 (ROMless) 16-Bit Mixed Signal Processor Preliminary Product Specification Z89121, Z89921 (ROMless) 16-Bit Mixed Signal Processor Preliminary Product Specification Z89320 16-Bit Digital Signal Processor Preliminary Product Specification Z89321 16-Bit Digital Signal Processor Advance Information Specification	DC-8299-01	3.00
Telephone Answering Device Databook (includes the following documents) Z89C65, Z89C66 (ROMless) Dual Processor T.A.M. Controller Preliminary Product Specification Z89C67, Z89C68 (ROMless) Dual Processor Tapeless T.A.M. Controller Preliminary Product Specification	DC-8300-01	3.00
Infrared Remote (IR) Control Databook (includes the following documents) Z86L06 Low Voltage CMOS Consumer Controller Processor Preliminary Product Specification Z86L29 6K Infrared (IR) Remote (ZIRC™) Controller Advance Information Specification Z86L70/L71/L72, Z86E72 Zilog IR (ZIRC™) CCP™ Controller Family Preliminary Product Specification	DC-8301-01	3.00

Z8 Product Specifications, Technical Manuals and Users Guides	Part No	Unit Cost
Z86E23 CMOS Z8 OTP Microcontroller Preliminary Product Specification	DC-2598-00	N/C
Z86C27/97 Z8 DTC™ Product Specification and Addendum	DC-2561-01	N/C
Z86127 Low-Cost Digital Television Controller Preliminary Product Specification	DC-2574-00	N/C
Z86227 40-Pin Low-Cost Digital Television Controller Preliminary Product Specification and Addendum	DC-3002-00	N/C
Z86C61/62/96 CMOS Z8 Microcontroller Preliminary Product Specification	DC-2587-00	N/C
Z86C93 CMOS Z8 ROMless Microcontroller Product Specification	DC-2508-03	N/C
Z86L70/71/72, Z86E72 Zilog IR (ZIRC™) Controller Family Preliminary Product Specification	DC-2620-00	N/C
Z88C00 CMOS Super8 ROMless Microcontroller Preliminary Product Specification	DC-2551-00	N/C
Z8604 NMOS Z8 8-Bit Microcontroller Preliminary Product Specification	DC-2524-03	N/C
Z8614 NMOS Z8 8-Bit MCU Keyboard Controller Preliminary Product Specification	DC-2576-00	N/C
Z86128 Closed-Captioned Controller Preliminary Product Specification and Addendum	DC-2570-01	N/C
Z86017 PCMCIA Adaptor Chip Advance Information Specification	DC-2643-0A	N/C
Z8 OTP CMOS One-Time-Programmable Microcontrollers Addendum	DC-2614-AA	N/C
asm S8 Super8/Z8 Cross Assembler User's Guide	DC-8267-05	3.00
Z8 Microcontrollers Technical Manual	DC-8291-02	5.00
Z86018 Preliminary User's Manual	DC-8296-00	N/C
Ditigal TV Controller User's Manual	DC-8289-01	3.00

Z8 Application Notes	Part No	Unit Cost
The Z8 MCU In Telephone Answering Systems	DC-2514-01	N/C
Z8602 Controls A 101/102 PC/Keyboard	DC-2601-01	N/C
The Z8 MCU Dual Analog Comparator	DC-2516-01	N/C
Z86C09/19 Low Cost Z8 MCU Emulator	DC-2537-01	N/C
Z8 Applications for I/O Port Expansions	DC-2539-01	N/C
Z86E21 Z8 Low Cost Thermal Printer	DC-2541-01	N/C
Zilog Family On-Chip Oscillator Design	DC-2496-01	N/C
Using the Zilog Z86C06 SPI Bus	DC-2584-01	N/C
Interfacing LCDs to the Z8	DC-2592-01	N/C
X-10 Compatible Infrared (IR) Remote Control	DC-2591-01	N/C
Z86C17 In-Mouse Applications	DC-3001-01	N/C
Z86C40/E40 MCU Applications Evaluation Board	DC-2604-01	N/C
Z86C08/C17 Controls A Scrolling LED Message Display	DC-2605-01	N/C
Z86C95 Hard Disk Controller Flash EPROM Interface	DC-2639-01	N/C
Timekeeping with Z8; DTMF Tone Generation; Serial Communication Using the CCP Software UART	DC-2645-01	N/C

Z80®/Z8000® CLASSIC FAMILY OF PRODUCTS

Z80®/Z180™/Z280®/Z8000® and Datacom Family	Part No	Unit Cost
Volume I Databook	**DC-2610-01**	**5.00**

Microprocessors and Peripherals

Discrete Z80® Family

Z8400/C00 NMOS/CMOS Z80® CPU Product Specification
Z8410/C10 NMOS/CMOS Z80 DMA Product Specification
Z8420/C20 NMOS/CMOS Z80 PIO Product Specification
Z8430/C30 NMOS/CMOS Z80 CTC Product Specification
Z8440/Z84C40 NMOS/CMOS Z80 SIO Product Specification

Embedded Controllers

Z84C01 Z80 CPU with CGC Product Specification
Z84C50 RAM80™ Preliminary Product Specification
Z8470 Z80 DART Product Specification
Z84C90 CMOS Z80 KIO™ Product Specification
Z84011/C11 PIO Parallel I/O Product Specification
Z84013/015 Z84C13/C15 IPC/EIPC™ Product Specification
Z80180/Z8S180 Z180 MPU Product Specification
Z80181 ZIO™ Controller Product Specification
Z280™ MPU Preliminary Product Specification

Serial Communications Controllers

Z8030/Z8530 Z-BUS® SCC Product Specification
Z80C30/Z85C30 SCC Product Specification
Z85230 ESCC™ Product Specification
Z80230 Z-BUS ESCC Product Specification
Z16C35 ISCC™ Product Specification
Z5380 SCSI Product Specification
Z53C80 SCSI Product Specification
Z85C80 SCSI/SCC Product Specification
Z16C30 USC™ Product Specification
Z16C32 IUSC™ Product Specification
Z16C33 MUSC™ Product Specification
Z16C50 DDPLL™ Product Specification

Technical Articles

Z80 Questions and Answers
Z180 Questions and Answers
SCC Questions and Answers
ESCC Questions and Answers
ISCC Questions and Answers

Additional Information

Superintegration Products Guide
Support Product Summary
Product Support
Military Qualified Products
Quality and Reliability
Literature Guide
Package Information
Ordering Information

Z80®/Z8000® CLASSIC FAMILY OF PRODUCTS (Continued)

Z80®/Z180™/Z280®/Z8000® and Datacom Family	Part No	Unit Cost
Volume II Databook	**DC-2622-01**	**5.00**

Microprocessors and Peripherals

Application Notes

Z80® Family Interrupt Structure
Using the Z80® SIO with SDLC
Using the Z80® SIO In Asynchronous Communications
Binary Synchronous Communication Using the Z80® SIO
Serial Communication with the Z80A DART
Interfacing 8500 Peripherals to the Z80®
Serial Clock Generation Using the Z8536 CIO
Timing in an Interrupt-Based System with the Z80® CTC
A Z80-Based System Using the DMA with the SIO
Interfacing the Z8500 Peripherals to the 68000
Z180™ Break Detection
The Z180™ Interfaced with the SCC at 10 MHz
Technical Considerations When Implementing LocalTalk Link Access Protocol
Using the Z84C11/C13/C15 in Place of the Z84011/013/015
Using SCC with Z8000 in SDLC Protocol
SCC in Binary Synchronous Communications
On-Chip Oscillator Design
Interfacing Z80® CPUs to the Z8500 Peripheral Family
Boost Your System Performance Using The Zilog ESCC™
Design a Serial Board to Handle Multiple Protocols
Using the Z16C30 USC Universal Serial Controller
Datacommunications IUSC/MUSC Time Slot Assigner
Integrating Serial Data and SCSI Peripheral Control on One Chip
A Fast Z80® Embedded Controller
Using the Zilog Datacom Family with the 80186 CPU

Questions and Answers

Z80® Questions and Answers
Z180™ Questions and Answers
SCC Questions and Answers
ESCC Questions and Answers
ISCC Questions and Answers

Additional Information

Classic Family
Datacom Products
Literature Guide

Z80/Z180/Z280 Product Specifications, Technical Manuals and Users Guides	Part No	Unit Cost
Z80 CPU Central Processing Unit Technical Manual	DC-0029-04	3.00
Z80 Family Programmer's Reference Guide	DC-0012-04	3.00
Z80 DMA Direct Memory Access Technical Manual	DC-2013-A0	3.00
Z80 PIO Parallel Input/Output Technical Manual	DC-0008-03	3.00
Z80 CTC Counter/Timer Circuit Technical Manual	DC-0036-03	3.00
Z80 SIO Serial I/O Technical Manual	DC-3033-01	3.00
Z80180 Z180 MPU Microprocessor Unit Technical Manual	DC-8276-03	3.00
Z280 MPU Microprocessor Unit Technical Manual	DC-8224-03	3.00
Z80181 Z181 ZIO™ Zilog I/O Controller Preliminary Product Specification	DC-2519-03	N/C
Z84C00 20 MHz Z80 CPU Central Processing Unit Preliminary Product Specification	DC-2523-02	N/C
Z84C50 Z80 RAM80 Z80 CPU/2K SRAM Preliminary Product Specification	DC-2498-01	N/C
Z80180/Z8S180 Z180 Microprocessor Product Specification	DC-2609-02	N/C
Z80182 Zilog Intelligent Peripheral (ZIP™)	DC-2616-02	N/C
Z380 Preliminary Product Specification	DC-3003-01	N/C

Z80/Z180/Z280 Application Notes	Part No	Unit Cost
Z180/SCC™ Serial Communications Controller Interface at 10 MHz	DC-2521-02	N/C
Z80 Using the 84C11/C13/C15 in place of the 84011/013/015	DC-2499-02	N/C
LocalTalk Link Access Protocol Using the Z80181	DC-2589-01	N/C
A Fast Z80 Embedded Controller	DC-2578-01	N/C

Z8000® MICROPROCESSOR FAMILY

Z8000 Product Specifications, Technical Manuals and Users Guides	Part No	Unit Cost
Z8000 CPU Central Processing Unit Technical Manual	DC-2010-06	3.00
SCC Serial Communication Controller User's Manual	DC-8293-02	3.00
Z8036 Z-CIO/Z8536 CIO Counter/Timer and Parallel Input/Output Technical Manual	DC-2091-02	3.00
Z8038 Z8000 Z-FIO FIFO Input/Output Interface Technical Manual	DC-2051-01	3.00
Z8000 CPU Central Processing Unit Programmer's Pocket Guide	DC-0122-03	3.00
Z5380 SCSI Small Computer System Interface Preliminary Product Specification	DC-2477-01	N/C
Z85233 EMSCC Enhanced Mono Serial Communication Controller Preliminary Product Specification	DC-2590-00	N/C
Z85C80 SCSCI™ Serial Communication and Small Computer Interface Preliminary Product Specification	DC-2534-02	N/C
Z16C30 CMOS USC™ Universal Serial Controller Preliminary Product Specification	DC-2492-03	N/C
Z16C30 USC Universal Serial Controller Preliminary Technical Manual	DC-8280-02	3.00
Z16C33 CMOS USC/MUSC™ Universal Serial Controller Technical Manual	DC-8285-01	3.00
Z16C33 CMOS USC/MUSC Universal Serial Controller Addendum	DC-8285-01A	N/C
Z16C32 IUSC™ Integrated Universal Serial Controller Product Specification	DC-2600-00	N/C
Z16C32 IUSC Integrated Universal Serial Controller Product Specification Addendum	DC-2600-00A	N/C
Z16C32 IUSC Integrated Universal Serial Controller Technical Manual	DC-8292-03	3.00
Z16C33 CMOS MUSC Mono-Universal Serial Controller Preliminary Product Specification	DC-2517-03	N/C
Z16C35 CMOS ISCC™ Integrated Serial Communication Controller Product Specification	DC-2515-03	N/C
Z16C35 ISCC Integrated Serial Communication Controller Technical Manual	DC-8286-01	3.00
Z16C35 ISCC Integrated Serial Communication Controller Addendum	DC-8286-01A	N/C
Z53C80 Small Computer System Interface (SCSI) Product Specification	DC-2575-01	N/C
Z80230 Z-BUS® ESCC Enhanced Serial Communication Controller Preliminary Product Specification	DC-2603-01	N/C

Z8000 Application Notes	Part No	Unit Cost
Z16C30 Using the USC in Military Applications	DC-2536-01	N/C
Datacom IUSC/MUSC Time Slot Assigner	DC-2497-02	N/C
Datacom Evaluation Board Using The Zilog Family With The 80186 CPU	DC-2560-03	N/C
Boost Your System Performance Using the Zilog ESCC Controller	DC-2555-02	N/C
Z16C30 USC - Design a Serial Board for Multiple Protocols	DC-2554-01	N/C
Integrating Serial Data and SCSI Peripheral Control on one Chip	DC-2594-01	N/C
Using a SCSI Port for Generalized I/O	DC-2608-01	N/C

MILITARY COMPONENTS FAMILY

Military Specifications	Part No	Unit Cost
Z8681 ROMless Microcomputer Military Product Specification	DC-2392-02	N/C
Z8001/8002 Military Z8000 CPU Central Processing Unit Military Product Specification	DC-2342-03	N/C
Z8581 Military CGC Clock Generator and Controller Military Product Specification	DC-2346-01	N/C
Z8030 Military Z8000 Z-SCC Serial Communications Controller Military Product Specification	DC-2388-02	N/C
Z8530 Military SCC Serial Communications Controller Military Product Specification	DC-2397-02	N/C
Z8036 Military Z8000 Z-CIO Counter/Timer Controller and Parallel I/O Military Electrical Specification	DC-2389-01	N/C
Z8038/8538 Military FIO FIFO Input/Output Interface Unit Military Product Specification	DC-2463-02	N/C
Z8536 Military CIO Counter/Timer Controller and Parallel I/O Military Electrical Specification	DC-2396-01	N/C
Z8400 Military Z80 CPU Central Processing Unit Military Electrical Specification	DC-2351-02	N/C
Z8420 Military PIO Parallel Input/Output Controller Military Product Specification	DC-2384-02	N/C
Z8430 Military CTC Counter/Timer Circuit Military Electrical Specification	DC-2385-01	N/C
Z8440/1/2/4 Z80 SIO Serial Input/Output Controller Military Product Specification	DC-2386-02	N/C
Z80C30/85C30 Military CMOS SCC Serial Communications Controller Military Product Specification	DC-2478-02	N/C
Z84C00 CMOS Z80 CPU Central Processing Unit Military Product Specification	DC-2441-02	N/C
Z84C20 CMOS Z80 PIO Parallel Input/Output Military Product Specification	DC-2384-02	N/C
Z84C30 CMOS Z80 CTC Counter/Timer Circuit Military Product Specification	DC-2481-01	N/C
Z84C40/1/2/4 CMOS Z80 SIO Serial Input/Output Military Product Specification	DC-2482-01	N/C
Z16C30 CMOS USC Universal Serial Controller Military Preliminary Product Specification	DC-2531-01	N/C
Z80180 Z180 MPU Microprocessor Unit Military Product Specification	DC-2538-01	N/C
Z84C90 CMOS KIO Serial/Parallel/Counter Timer Preliminary Military Product Specification	DC-2502-00	N/C
Z85230 ESCC Enhanced Serial Communication Controller Military Product Specification	DC-2595-00	N/C

GENERAL LITERATURE

Catalogs, Handbooks and Users Guides	Part No	Unit Cost
Superintegration Shortform Catalog 1992	DC-5472-10	N/C
Superintegration Products Guide	DC-5499-06	N/C
ZIA™3.3-5.5V Matched Chip Set for AT Hard Disk Drives Datasheet	DC-5556-01	N/C
Zilog Hard Disk Controllers - Z86C93/C95 Datasheet	DC-5560-01	N/C
Zilog Infrared (IR) Controllers - ZIRC™ Datasheet	DC-5558-01	N/C
Zilog Intelligent Peripheral Controller - ZIP™Z80182 Datasheet	DC-5525-01	N/C
Zilog Digital Signal Processing - Z89320 Datasheet	DC-5547-01	N/C
Zilog Datacommunications Brochure	DC-5519-00	N/C
Zilog Digital Signal Processing Brochure	DC-5536-01	N/C
Quality and Reliability Report	DC-2475-10	N/C
The Handling and Storage of Surface Mount Devices User's Guide	DC-5500-02	N/C
Support Products Summary	DC-2545-03	N/C
Universal Object File Utilities User's Guide	DC-8236-04	3.00
Zilog 1991 Annual Report	DC-1991-AR	N/C
Microcontroller Quick Reference Folder	DC-5508-01	N/C